
6,- DM öS 50,- sfr 6,-
CPC
Amstrad
INTERNATIONAL
CPC • PCW JOYCE
10/11
Hardware-Projekte
Heiß
Programme für Sie
Brandaktuell
PCW

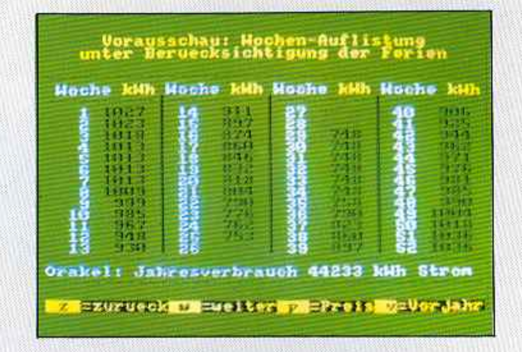
Inhalt:

für den privaten Haushalt:
- Kontenverwaltung
- Haushaltsbilanzen
- Orakel: Hochrechnung effektiver Verbraucherkosten
- Lohnsteuerberechnung
- KFZ-Kosten
- Belegverwaltung
- Bankformulare
- Wertpapierverwaltung
- Komfortable Währungsumrechnung
- Tilgung
- Komplette Mini-Tabellenkalkulation

für kleinere Business-Anwendungen:
- Rechnungsschreibung
- Pecunia
- Bestellung und Bestand
- Superplan

Der Finanz-Fachmann

Ein CPC-Mega-Sampler, voll mit praktischen Anwenderprogrammen rund ums liebe Geld



DM 49,-*



Lernen mit Spaß

Sicherlich liegt es Ihnen auch am Herzen, daß die Sprößlinge eifrig und effektiv lernen und nicht nur stur büffeln.
Der Pädagoge Berthold Freier hat mit den Programmdisketten "Lernen mit Spaß" 1 und 2 ein Lerninstrument geschaffen, das in dieser Art einmalig ist. Auf den Disketten befinden sich zahlreiche Spielprogramme, die Ihren Kindern spielerisch das Verständnis für Zahlen, Text und Farben vermitteln. Auch das Allgemeinwissen wird ausreichend gefördert.
Für Kinder von 4 bis 12 Jahren wird durch "Lernen mit Spaß" 1 und 2 der CPC-Computer zum lehrreichen Freund.
Machen Sie Ihren Kindern mit "Lernen mit Spaß" eine Freude.

GLEICHE SCHNAPPEN

Herr Blattroll?
Der wohnt in einer
tieferen Nummer !

Lernen mit Spaß 1 **DM 24,95***

Lernen mit Spaß 2 **DM 24,95***

Joyce-Highlights: Software-Perlen für Ihren Joyce/PCW

JOYCE-Highlights I: "Anwender"

Viele nützliche Arbeitserleichterungen

Spooler: Druckerspooler (256 und 512 KByte) • CAT: zeigt Disketteninhalt sortiert, auch versteckte Dateien • Loco2Pro: Konvertierung von LocoScript nach Prowort File • Rescue: Retten gelöschter Daten • Akte: Etikettenprogramm • Kalender: Feiertage, persönliche Daten • Drucken List V2.1: Listingdrucken und Kommentare • Funktionstasten: zeigt die Belegung auf allen fünf Ebenen • Für immer und ewig: Komprimieren und Entpacken von Dateien • Vario: Hardcopy auf (fast) allen Druckern • JOYCE-Tools: Benutzeroberfläche für Dateibearbeitung

nur DM 29,-*

JOYCE-Highlights II: "Spiele"

Für jeden etwas dabei. 14 ausgesuchte Spiele als Programmpaket – vom Kinderspiel über Unterhaltung bis zum kniffligen Denkspiel

- Gobang
- Pingo
- Backgammon
- Mensch, ärgere dich nicht
- Lokomotive
- Rubik's Clock
- Industriemanager
- Jackpot
- Poker
- Fruit
- Master Mind
- Ölmanager
- Verbindungen
- Q-Bert

nur DM 29,-*

Unabhängig von der Anzahl der bestellten Programme berechnen wir für das Inland 4,— DM bzw. für das Ausland 6,— DM Porto und Verpackung. Bitte benutzen Sie die Bestellkarte.

DMV-Verlag • Postfach 250 • W-3440 Eschwege





Olympisches Gold für Abtipp-Athleten!

Wieder einmal haben wir uns für Sie die Finger heißgetippt. Seite für Seite wuchsen die Beiträge zu ansehnlichen Manuskriptbergen heran, und das Klappern unserer Tastaturen bildete die mitreißende Hintergrundmusik. Aber jetzt ist es soweit: Die neue Ausgabe der CPC International "steht" – und ein gutes Dutzend hervorragender Programmlistings wartet nun seinerseits darauf, Ihre Finger zum Glühen zu bringen und den Weg in Ihren Computer zu finden.

Viele von Ihnen, liebe Leser, werden nun wieder zur Höchstform auflaufen, wenn es darum geht, die für Sie attraktivsten Programme abzutippen. Da gilt es, weder Finger noch Tastatur zu schonen:

"Soll das Programm den Tipper loben,
will stolz auf Kilobytes er schauen,
so heißt's: die Hand mit Fleiß erhoben
und in die Tasten forsch gehauen!"

Es ist unverfälschter Sportsgeist, der den echten Abtipper auszeichnet. Zielbewußtsein, Zähigkeit und Durchhaltevermögen sind seine Tugenden.

Die jüngst beendeten Olympischen Spiele in Barcelona haben den deutschen Athleten einen reichen Medaillensegen gebracht. Wieder gab es einige neue Disziplinen, die im olympischen Rahmen vorher nicht aufgetaucht waren – wie beispielsweise Badminton. Der Trend geht zu einer noch stärkeren Öffnung für bislang ausgeschlossene Sportarten.

Und wenn dann demnächst sogar Augenzwinkern und Halmaspielen zu olympischen Disziplinen geworden sind, wird man vielleicht auch dem Abtippsport endlich die längst fällige olympische Anerkennung zollen. Ich meine, es ist höchste Zeit, das Abtippen von Listings – ob als Marathon-Programmeingabe oder als Trickkisten-Kurzstreckensprint – als das gelten zu lassen, was es ist: eine echte sportliche Leistung.

Wenn dann Abtippleistungen tatsächlich mit olympischem Gold belohnt werden, dürften die CPCler und PCWler schnell zu den Favoriten der Nation werden. Gegenüber den satten, untrainierten Benutzern anderer Systeme, die von softwaremäßiger Fertignahrung verweichlicht sind, haben die Freunde der 8-Bit-Amstrad-Computer einen ganz klaren Vorteil: Sie sind fit! Ihre gute Form verdanken sie dem regelmäßigen Abtippen von Listings, dem selbständigen Verändern von Programmen, dem Einarbeiten eigener Lösungsansätze. Und damit Sie im Training bleiben, liefern wir Ihnen hiermit wieder kilobyteweise Kraftnahrung – auf daß die nächsten Olympischen Spiele ein Triumph der Abtipp-Athleten werden.

Natürlich kommen auch weniger sportliche Naturen bei uns auf ihre Kosten: Immerhin kann man unsere Software ja auch auf Diskette bekommen ...

Mit einem herzhaften "An die Tasten – fertig – los!" grüßt Sie Ihr

Peter Schmitz

Peter Schmitz, Chefredakteur

INHALT

BERICHT

KNOW-HOW

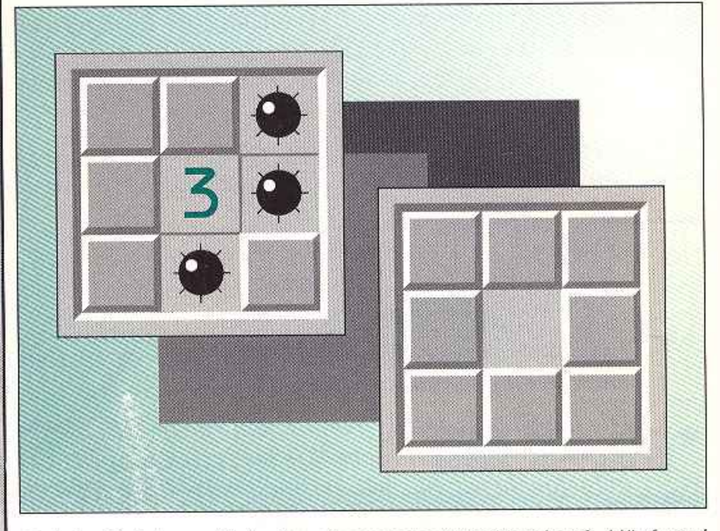
TIPS & TRICKS

REVIEW



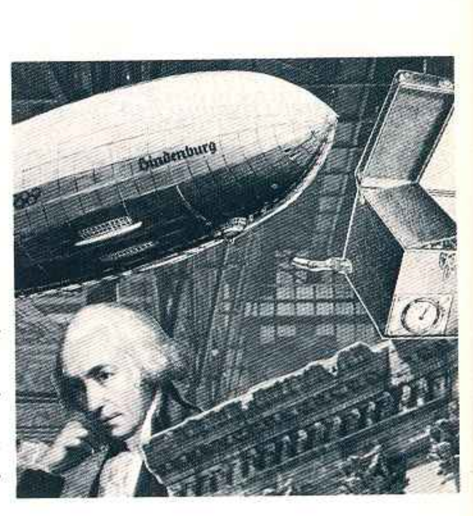
Cracker, Raubkopierer, Software-Piraten: Es gibt sie seit der ersten Stunde der Heimcomputer. Sie verstecken sich vor der Justiz, schädigen Händler und Hersteller. Sie haben ihre eigenen Verteiler-Organisationen und ihre eigene Moral entwickelt. Einer der ehemals aktivsten Programmknacker der deutschen CPC-Szene hat ausgepackt – lesen Sie unser Interview!

Seite 8



Bei den Spielen geht der Trend weg von eintönigen Lauf-, Hüpf- und Baller-Orgien. Mehr und mehr sind Programme gefragt, die Spannung und Kopftraining miteinander verbinden. Wenn dann auch noch attraktive Grafik dazukommt, stimmt die Mischung – wie bei "Think", unserem hochkarätigen Kombinationsspiel zum Abtippen.

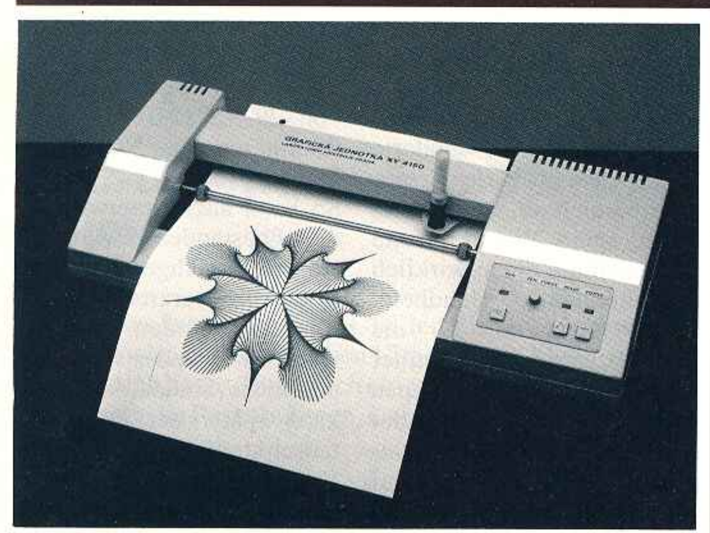
Seite 42



"Keine Zukunft ohne Vergangenheit!", haben kluge Leute gesagt. Auf jeden Fall gibt es keine Allgemeinbildung ohne geschichtliche Daten. Der Umgang damit braucht jedoch nicht in Tabellenkritzelei und grüblerisches Herumblättern auszuarten: Unsere Geschichtsdatenbank "History" macht das Lernen und Abrufen historischer Ereignisse zum Vergnügen.

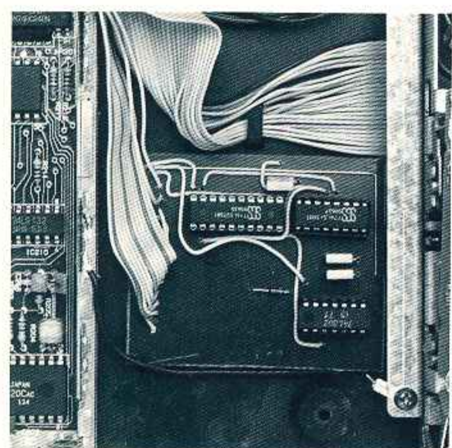
Seite 36

Oktober/Nov. '92



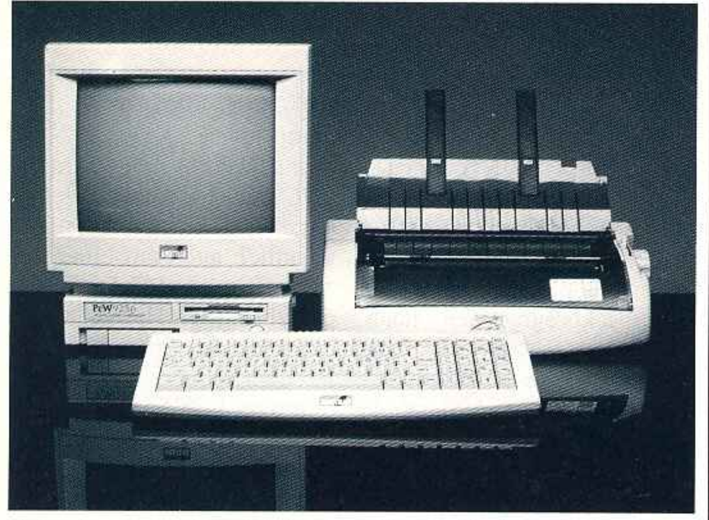
CPC-Grafikfreunde haben lange auf einen Stiftplotter gewartet. Wir haben einen gefunden! Der unverschämt preisgünstige XY 4150 aus Prag bringt bereits maßgeschneiderte Treibersoftware für den CPC mit. Wir setzen noch eins drauf und veröffentlichen das erste Vektorgrafikprogramm für den neuen Plotter.

Seite 92



Viel Gelegenheit zum Basteln gibt's auch diesmal wieder: Wie wäre es mit einem CP/M-Blitzstart auf Knopfdruck – ohne Ladezeiten? Auch das CPC-"Meßlabor" ist nicht ohne: Es erlaubt Ihnen das Testen von TTL-Chips sowie das Messen von Frequenzen und Kapazitäten. Wir präsentieren Ihnen ausführliche Bauanleitungen und sorgfältig entwickelte Platinenlayouts.

Seite 44



Schon lange angekündigt, fand das jüngste Kind aus der Joyce-Familie jetzt endlich den Weg zu den deutschen Käufern. Der neue PCW 9256 ist so kompakt, daß er auch auf der kleinsten Schreibtisch-Lichtung Platz findet.

Seite 100

ANGELESEN

PCW

CP/M

CPC-PROGRAMME

SERVICE

RUBRIKEN

Neuer CPC-Händler

Frisch gegründet präsentiert sich ein neuer Softwareversand, dessen Schwerpunkt auf CPC-Software liegt. Brostedt Computertechnik bietet nicht nur topaktuelle Spiele, sondern auch ältere CPC-Hits an. Wie man uns mitteilte, soll es auch eine eigene Entwicklung für neue CPC-Programme geben.
Info: Brostedt Computertechnik
Harkortstraße 54
4690 Herne 2
Tel.: 02325/60865

Hexenküche Teil 3 in Arbeit

Wie uns aus Insiderkreisen berichtet wurde, plant das französische Softwarehaus Titus die Veröffentlichung einer neuen Episode der "Hexenküche" für den CPC. Das Vorgängerspiel war gegen Mitte der achtziger Jahre ein absoluter Hit auf 8-Bit-Computersystemen.
Die Hexe mit dem Knickhut, die murmelnd in ihrem großen Kessel rührte, wurde zum Markenzeichen für ein ganzes Genre humorvoller Geschicklichkeitsspiele mit Comic-artiger Grafik.
Programmieren wird das neue Spiel kein anderer als Elmar Krieger, der sich schon durch das Programm ZAP'T' BALLS sowie durch diverse Demos in Deutschland einen Namen machen konnte.

Computerpflege statt Autowaschen

Die Picco Computerdienstleistungen GmbH bietet jetzt einen recht außergewöhnlichen Dienst an. So wird von Picco jedes PC-System mit biologisch abbaubaren Computer-Reinigungsmitteln gesäubert. Wie Joachim Schmidt, Geschäftsführer bei Picco, bemerkt, sind nahezu 90 Prozent aller Computerausfälle auf Verunreinigungen zurückzuführen. Daher plädiert die Picco GmbH an ihre Kunden, eine regelmäßige PC-Wäsche der regelmäßigen Autoinspektion gleichzustellen.

Info:
PICCO Computerdienstleistungen GmbH, Ridlerstraße 11, 8000 München 2,
Tel.: 0 89/50 20 09 31

Zap'T'Balls-Verlosung

In Verbindung mit Firma der NoName EDV-Service verlost die CPC Amstrad International drei Originalspiele "Zap'T'Balls Advanced Edition" sowie sieben "Zap'T'Balls"-T-Shirts. Was Sie machen müssen, um einen dieser Preise zu ergattern? Ganz einfach: Schicken Sie einfach eine (gut lesbare) Postkarte an:
CPC International
Kennwort: Zap'T'Balls
Fuldaer Straße 6
3440 Eschwege
Wie immmer ist der Rechtsweg ausgeschlossen. Mitarbeiter des DMV-Verlags sowie der NoName EDV-Service Gbr können nicht teilnehmen.

Amstrad Notepad NC100

Mit dem Prädikat "User-Friendly" hat die Firma Amstrad ihre neueste Errungenschaft gleich schon einmal selbst versehen. Obwohl vielerseits befürchtet, wanderte Amstrad nicht aus dem Computerbereich ab, sondern ergänzt seine Produktpalette nun noch um einen wirklich ansehnlichen Handheld. Dieser Computer für die Manteltasche erscheint unter der Bezeichnung "Notepad NC 100" und ähnelt stark der Sinclair-Entwicklung "Cambridge Z88". Basierend auf einem Z80-Chip, arbeitet das Gerät in der Grundausstattung mit 64 kByte batteriegepuffertem Speicher. Dieser kann jedoch durch eine Erweiterungskarte auf 1 MByte aufgerüstet werden. Das Gewicht von nur einem Kilogramm sowie die Ausmaße von 297x27x210 mm prädestinieren den NC100 als Reisebegleiter.
So kann auch unterwegs in Adreßbeständen gewütet, im Kalender nachgeschaut und in der Textverarbeitung geschrieben werden. Über jeweils ein serielles sowie ein paralleles Standardinterface kann später der Datenaustausch stattfinden.
Einziges Manko, die Frage, ob das Gerät auch in Deutschland vertrieben werden soll, ist noch offen.

Info: Amstrad GmbH

LocoScript PC: Handbücher jetzt in Deutsch

Ab Oktober 1992 beginnt die Auslieferung der deutschen Dokumentation zu LocoScript PC. Jeder registrierte LocoScript-PC-Anwender kann so das LocoScript-Nachschlagewerk zu einem Preis von 119,70 DM erwerben. Neben dem Buch liegt der Sendung dann noch eine Diskette mit Lehrbeispielen bei.
Info: Wiedmann Unternehmensberatung & Handel
Korbiniansplatz 2
8045 Ismaning

Neue PD-Serie bei DMV

Der Daten und Medienverlag, Eschwege wird jetzt speziell für den CPC als auch für den PCW eine eigene PD-Serie aufbauen. Auf der bisher verfügbaren Diskette für den CPC befinden sich neben den gängigsten CP/M-Dateipackprogrammen noch das Spiel Croco Magneto sowie das deutsche Disketten- und Dateikopierprogramm CRIME.
Aber auch die PCW-PD-Disk kann jede Menge interessante Programme vorweisen. Befinden sich doch auf dieser Disk äußerst beeindruckende Programme aus vorangegangenen Ausgaben der CPC. Alles Public Domain!
Wer selbst ein Programm für CPC oder PCW geschrieben hat, kann dies selbstverständlich auch auf einer der zukünftigen PD-Disketten unterbringen. Hierzu sollte lediglich das Programm mit Beschreibung auf einen Datenträger kopiert und an folgende Adresse geschickt werden:
DMV
CPC/PCW Public Domain Service
Fuldaer Str. 6
3440 Eschwege

Auf Schusters Rappen

CPC-Userparties in Ulm und Reims


Das Marabu Meeting 1992: Gar nicht so leicht, alle auf einem Foto unterzubringen

Auch dieses Jahr trafen sich CPCler aller Länder zum Fachsimpeln. In der Nähe von Ulm fand ein Meeting statt, das es in sich hatte. Aber auch ein paar Tage später in Reims ging die Post ab. Feten, Labern und Coden war angesagt.

Ulm, eine Stadt, die nicht nur für ihre großen Ausbildungszentren bekannt ist, stand diesmal auf dem Plan der CPC-Szene-Freaks. Grund für diesen Eintrag im Terminplan? Marabu gab eine gewaltige Party. Sie fand zwar nicht im Ulmer Münster statt, Platz war trotzdem mehr als genug vorhanden. Den Partygästen stand eine ganze Etage eines Mehrfamilienhauses zur Verfügung. Insgesamt trafen sich über 20 Coder aus der internationalen Szene.
Zeitlich passend zur Party wurde die neue CI, ein Fanzine der deutschen CPC-Szene, herausgebracht. Crackers International gab wie immer Grund zum Fachsimpeln. Als dann noch die CPC FASTLOADER, ein deutsches Diskettenmagazin präsentiert wurde, war der Kommunikationsdrang der Partygäste nicht mehr zu bremsen.
Absoluter Höhepunkt dieses Usertreffens war jedoch die Erstpräsentation der Advanced Edition von Zap'T'Balls. Obwohl die Urversion dieses Games schon von der DATA-BOX zu der Ausgabe 4/5'92 bekannt war, konnte Zap'T'Balls durch neue Levels und noch ansprechendere Grafiken überzeugen. Das Ball-Fieber griff um sich.
Nach Meinung aller Beteiligten war die Verpflegung auf dieser Party bisher am besten. Stand doch in der Wohnetage neben normalen Wohnräumen auch ein reichlich gefüllter Kühlschrank in der Küche zur Verfügung.

Die Franzosen rufen

Genau eine Woche nach den Geschehnissen in Ulm riefen die Franzosen zum großen CPC-Klamauk. Etwa 500 Kilometer von Ulm entfernt, versammelte man sich in der Stadt Reims. Für eine Gebühr von umgerechnet 30,– DM konnte dort jeder mitmischen. Platzmangel, darunter hatte auch hier wirklich keiner zu leiden. Für etwa fünfzig CPC-Freaks, wurde eigens eine Halle angemietet. Reims, Ort der Veranstaltung, lockte nicht nur durch Prominente aus der CPC-Szene. So waren auch unsere Kollegen von der CPC Attack und der Amstrad Action aus England angereist. Die französische CPC-Presse repräsentierte die Amstrad Cent Pour Cent.
Neben dem üblichen Fachgesimpel wurde auch hier wieder munter gecoded und in anderen Demos gehackt. Neben unzähligen CPCs war auch ein Außenseiter, ein SEGA Mega Drive, vorhanden.
Auch dieses Treffen brachte wieder neue Demos hervor. So wurden zum Beispiel von Odiesoft alle Parts der in Ulm angefangenen Castle-Demo verbunden.

Mike Behrendt/rs



Meeting in Frankreich: CPCs en masse. Wo sollen denn da die Besucher noch Platz finden?


Die deutsche Ecke des Frankreich-Meetings. Warum können die Deutschen eigentlich keine Ordnung halten?



"Black Mission" packt aus

Ein Personality-Bericht der etwas anderen Art: Wir sprachen mit einem CPC-Cracker

18 Jahre, Abi im Frühjahr 1992, jetzt angehender Bankkaufmann: Unter dem Namen "Black Mission" kennt ihn die internationale CPC-Raubkopiererszene. Er ist Deutschlands Cracker Nummer 1, ein rotes Tuch für Software-Kaufleute und EDV-Fahnder. Über 150 Programme hat "Black Mission" geknackt. Nach seiner "Bearbeitung" ist das Kopieren eines Spiels, das im Laden 50 DM oder mehr kosten würde, eine Sache von wenigen Sekunden.

Im Herbst 1991 machte eine Nachricht wie ein Lauffeuer die Runde durch die illegale Cracker- und Kopiererszene: "Black Mission ist ausgestiegen." Dann im Frühjahr 1992, nach dem Abi, ein Auftritt auf der CeBIT – und "Black Mission macht weiter". Heute crackt er im kleinen Maßstab, wenn er mal ein neues Programm bekommt, das noch keiner vorher "geschafft" hat.

In unserem Interview spricht er über Beweggründe, Moral und die Angst vor der Polizei.

***Redaktion CPC:** Wann hast du das erste Mal ein Programm geknackt?*

"Black Mission": Das war im Dezember 1988, und zwar "Cybernoid II".

***Red.:** Wie kamst du auf die Idee, ein Cracker zu werden?*

BM: Ein Tauschpartner hatte sich das Original gekauft und mir eine Kopie geschickt – mit der Bitte, es zu knacken. Ich machte mich dann zusammen mit einem Freund von mir, der schon öfter gecrackt hatte, daran, das Spiel zu entschützen.

***Red.:** Das war also das erste Mal. Wie hat es sich dann weiterentwickelt? Man wird ja nicht 150mal hintereinander gefragt, ob man etwas knacken will, oder?*

BM: Mein Tauschpartner wollte das Spiel möglichst schnell zurückhaben, was dann auch so innerhalb von zwei Tagen geschah. Darüber war er recht erfreut und fragte, ob ich das häufiger für ihn machen wollte, weil er sich recht oft Originale kaufen würde. Von da an hat er mir öfter Originale geschickt. Ich habe dann versucht, auch woanders noch welche aufzutreiben. Mit der Zeit wurden das dann immer mehr Leute und auch immer mehr Spiele.

***Red.:** Was bringt dich dazu, deine Zeit in das Knacken von Spielesoftware zu investieren? Wie wichtig ist es dir dabei, daß dein Name – dein Pseudonym – bekannt wird?*

BM: Anfangs hat das eine sehr große Rolle gespielt. Als ich mein erstes Spiel gecrackt habe, beherrschte ich nur Grundzüge von Assembler. Das reichte zum Knacken aus, aber noch nicht zum Schreiben von Demos. Dazu muß ich sagen, daß das Cracken meiner Meinung nach ziemlich einfach ist.

Und ich wollte, das gebe ich offen zu, bekannt werden. Wer will das nicht? Da habe ich es halt mit Cracken versucht, was recht gute Erfolge versprach. Ich wollte zuerst in Deutschland bekannt werden, dann auch im Ausland. Mit der Zeit bekam ich dann internationale Kontakte in Frankreich, Spanien und sogar Griechenland. So wurde ich dann auch im Ausland bekannt. Aber so ab '90 oder '91 kam es mir darauf nicht mehr so sehr an. Da wurde es dann eher zur Routine.

***Red.:** Wie lange brauchst du, um ein Spiel zu knacken?*

BM: Das kommt ganz auf das Spiel an – ob das jetzt mit oder ohne Nachlader läuft und welchen Kopierschutz es hat. Das dauert, grob gesagt, zwischen einer halben Stunde und sechs Stunden, mit Laderoutine und Scrolltext.

***Red.:** Cracken und Raubkopieren ist illegal. Hast du keine Angst, von der Polizei erwischt zu werden?*

BM: Ich bin schon einmal erwischt worden. Die Polizei hatte meine Adresse bei einem Tauschpartner gefunden. Sie kamen aber erst ein Dreivierteljahr später und haben dann alles durchsucht. Ich hatte vorgesorgt und die Disketten verschwinden lassen. Nicht auf Grund eines Tips, sondern einfach als Sicherheitsmaßnahme. Somit haben die Cops außer Leerdisketten nichts gefunden.

Sie haben dann zwar den Computer trotzdem mitgenommen, ihn aber nach sechs Wochen wieder zurückgebracht. Die Gefahr war und ist natürlich da. Ich versuche aber, ihr entgegenzuwirken, indem ich die Disketten nicht in meinem Zimmer unterbringe. In den letzten Jahren habe ich verstärkt versucht, die Tauschpartner ins Ausland zu verlegen. Dort interessiert es nämlich fast niemanden, wer was kopiert. Vor allem die Polizei nicht. Es gibt zwar auch dort private Anti-Raubkopier-Organisationen, aber die haben keinen Einfluß. Adressen und vor allem Kopien zu besitzen, ist also überhaupt kein Risiko.

***Red.:** Ich stelle mir vor, daß es zumindest beim ersten Mal eine gewisse Überwindung ist, etwas Illegales zu machen.*

BM: Zuerst war da der Anreiz, bekannt zu werden. Daß es illegal war, war eigentlich zweitrangig. Ich habe mir gedacht: "Bekannt werden – ja! Illegal – nun gut. Aber mich werden sie schon nicht erwischen. 8-Bit-Computer interessieren sowieso keinen mehr, die packen nur Leute auf dem Amiga oder auf dem Atari." Das war dann aber ein Trugschluß.

***Red.:** Was würdest du denn sagen, wenn du selbst ein Programm schreiben würdest und dieses dann fleißig geknackt und weiterkopiert würde?*

BM: Da würde ich mich wahrscheinlich schon ein bißchen ärgern. Ich muß aber dazu sagen, gerade in letzter Zeit habe ich mir von guten Programmen meist das Original zugelegt. Und früher habe ich nicht groß nachgedacht. Mir war es ziemlich egal, was mit den Programmierern passiert. Da hätte ich wohl gesagt: "Die Programme sind mit wenig Aufwand vom Spectrum rübergezogen worden, da hat man auch nicht viel investiert. Dementsprechend können die Softwarefirmen auch keinen so riesigen Verlust machen, wenn die Spiele nicht gekauft werden." Klar, daß das auch nicht ganz richtig und nicht ganz logisch ist. Eine richtige Rechtfertigung für das Cracken gibt es nicht und wird es auch nie geben. Es wird immer illegal bleiben.

***Red.:** Auch wenn die Konvertierungen vom Spectrum ziemlich leicht sind, ziehen sich in letzter Zeit immer mehr Firmen vom CPC zurück, weil es sich nicht mehr lohnt. Wenn die Raubkopiererei so weitergeht wie bisher, wird damit der CPC noch schneller zu Grabe getragen ...*

BM: Ich würde sagen, das ist nur teilweise richtig. Schaut man sich den Markt in Frankreich an – dort boomt der CPC seit eh und je, und dort wird bestimmt nicht weniger kopiert. Deshalb kann ich mir nicht vorstellen, daß das der Hauptgrund ist. Vielleicht ist die Motivation in Deutschland anders, daß hier noch weniger Programme gekauft werden, obwohl die Games in Frankreich noch mehr kosten? Ich glaube, daß die Raubkopiererei einer von mehreren Gründen ist, daß Softwarefirmen zugrundegehen, aber nicht der Hauptgrund. Ich gebe allerdings zu, daß es den Markt verkleinert.

***Red.:** Dann sind da noch die Leute, die an der Raubkopiererei kräftig verdienen. Die kassieren, sagen wir mal, zehn Mark pro Diskette. Wie stehst du dazu?*

BM: Ja, die gibt es. Vor allem auf dem Amiga und Atari. Auf dem Amstrad ist mir das nur in ganz wenigen Fällen bekannt, vor allem auch nicht zu solchen Preisen. Es ist zwar keine Rechtfertigung, aber manche mußten eben irgendwie ihre hohen Portokosten und den Zeitaufwand decken. Davon will ich mich nicht ausschließen, denn sonst hätte ich gut und gerne 100 bis 150 DM pro Monat für Portokosten und ähnliches aufbringen müssen.

***Red.:** Inzwischen müßtest du doch so viel Ahnung von Assembler haben, daß du selbst eine Demo oder ein Spiel programmieren könntest. Das würde allen viel mehr nützen, und du würdest dadurch noch weiter bekannt werden.*

BM: Ich habe im letzten Jahr weniger gecrackt als vorher und eher etwas in Richtung der Demoschreiberei getan. Einige Demos, darunter einige noch unveröffentlichte, habe ich bereits geschrieben. Aber ich stelle immer noch fest, daß ich mit guten Programmierern, wie etwa denen von Logon System, nicht mithalten kann. Warum weiß ich nicht. Vielleicht beschäftigen die sich mehr mit dem CPC. Und – ein Spiel zu programmieren, ist noch ein ganz anderes Kaliber als eine Demo. Ich glaube, ich wäre dazu auch nicht in der Lage. Demos mittlerweile schon, auch ordentliche, aber nichts Überragendes. Da fehlen mir einfach noch Kenntnisse über "illegale" Befehle und ähnliches.

***Red.:** Vorhin hast du gesagt, du brauchst zum Knacken eines Spieles eine halbe Stunde bis sechs Stunden. Eine Zeitlang war dein wöchentlicher Spieleausstoß ganz beachtlich. Da geht dann doch ziemlich viel Zeit dafür drauf, so daß kaum noch etwas anderes Platz hat ...*

BM: Das ist richtig. Da gab es die Schule und nachmittags den Computer. 1989, 1990 habe ich oft so zwei bis drei Sendungen am Tag bekommen. Die habe ich dann bearbeitet und eventuell ein Spiel gecrackt. Ich hatte wirklich nicht mehr viel Freizeit, da ist durchaus etwas verlorengegangen. Dazu muß ich aber sagen, daß das genauso für Demo-Coder zutrifft. Die sitzen teilweise auch 24 Stunden nonstop am Computer, um eine einzige neue Routine zu entwickeln. Also ist der Freizeitverlust nicht nur beim Cracken zu verzeichnen. Inzwischen spielen meine Hobbies wieder eine stärkere Rolle: Sport, Fußball, das andere Geschlecht, Musik, Discos und so weiter – aber der Computer natürlich auch immer noch.

***Red.:** Wenn es irgendwann keine neuen Spiele mehr für den CPC gibt und du dir vielleicht einen anderen Computer zulegst, fängst du dann wieder mit Cracken und Demoschreiben an, oder ist dieses Kapitel dann abgehakt?*

BM: Mit Cracken fange ich dann garantiert nicht mehr an. Mit Demoschreiben ja, wenn ich mich mit der Programmiersprache zurechtfinde. Beim Cracken ist das jetzt schon so: Ich cracke ein Game, wenn ich Lust dazu habe – und wenn nicht, dann eben nicht. Wenn ich Originale bekomme, verspüre ich nicht mehr unbedingt den Zwang, die zu cracken. Das war noch vor zwei Jahren ganz anders.

***Red.:** Wenn du jetzt also noch etwas crackst, dann einfach deshalb, weil man es von dir erwartet?*

BM: Ja, so kann man das sehen. Sonst heißt es ja wieder wie im Herbst '91 "Black Mission ist ausgestiegen".

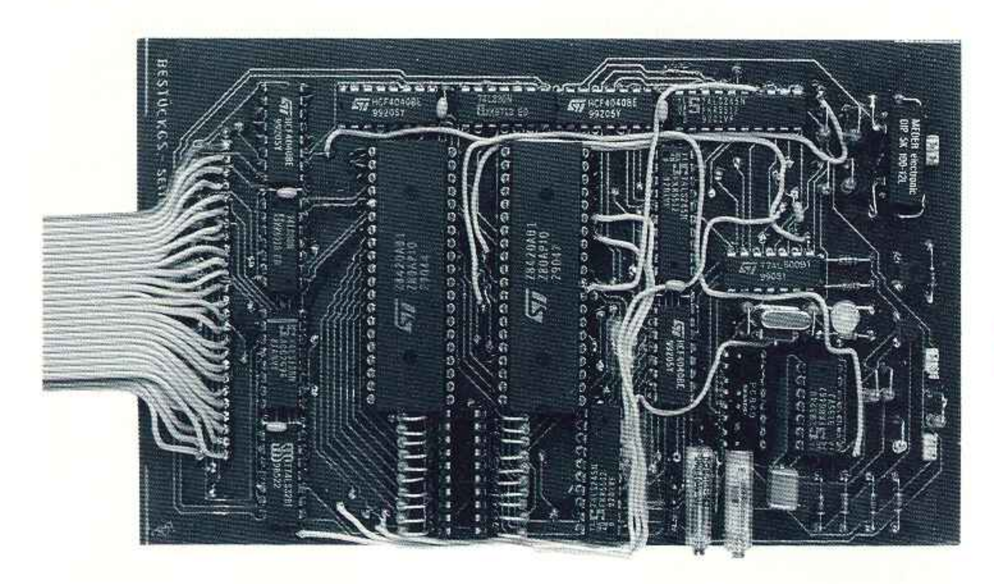
***Red.:** Ich danke dir für das Interview.*

Andreas Lober/rs

Stichwort "Cracken"

Vom "Cracken" oder "Knacken" spricht man, wenn jemand den Kopierschutz einer Original-Software entfernt und eine lauffähige Kopie dieses Originals ohne Kopierschutz herstellt. Das Ziel dieser Tätigkeit besteht darin, daß sich das "gecrackte" Programm mit beliebigen Kopierprogrammen vervielfältigen und so über einen Verteiler-Kreis verbreiten läßt. Der "Cracker" spielt dabei mit dem Feuer: Wenn er sich einen "wasserdichten" Verteiler-Ring aufgebaut hat, kann er durch "Abonnementsgebühren" seiner Abnehmer unter Umständen viel Geld einnehmen. Da jedoch die Veränderung wie auch die nichtlizenzierte Vervielfältigung und Weitergabe Copyright-geschützter Software nach bundesdeutschem Recht strafbar sind, hat ein erwischter Cracker nichts zu lachen. Nachdem zunächst seine Computeranlage beschlagnahmt worden ist, winkt ihm abhängig von seinem Lebensalter und dem Umfang seiner Aktivitäten in der Regel eine empfindliche Geld- oder unter Umständen sogar eine Haftstrafe.

"Cracker" sind selten Einzelgänger; sie arbeiten meist in Teams oder Clubs. Ein Cracker verewigt in der Regel seinen Decknamen in jedem Programm, das durch ihn seines Kopierschutzes beraubt wurde – manchmal in Form eines Lauftextes ("Scroller"). Oft leisten die Cracker-Teams auch als "Coder" selbst hervorragende Programmierarbeit, indem sie geknackte Spielprogramme mit "Intros" (grafisch aufwendigen Vorspännen) versehen oder atemberaubende Demos herstellen. Es gibt jedoch auch Gruppen, die sich nur auf die eigene kreative Programmierarbeit beschränken – so etwa "Logon System", eine französische Gruppe von Codern, die durch ihre bestechenden Demos (etwa "THE DEMO") bekannt wurde.



Der CPC als Meßlabor

Selbstbau eines Meßwandlers

Sind Sie Elektronikbastler? Dann haben Sie sicherlich ein Vielfachmeßgerät. Aber können Sie damit auch Frequenzen bis 16 Mhz messen, TTL-Schaltkreise auf ihre korrekte Funktion kontrollieren oder Kondensatoren nachmessen? Nein. Mit unserem Hardwareselbstbau für den CPC wird das alles möglich.

Wie das Echo auf unsere letzten Ausgaben zeigte, stehen Bauanleitungen bei unseren Lesern ganz hoch im Kurs. Deshalb ist auch in dieser Ausgabe die Hardwarebastelei wieder ein Schwerpunkt. Damit Ihnen in Zukunft die Inbetriebnahme Ihrer Eigenbauten besser gelingt, verwandelt die im folgenden beschriebene Hardware Ihren CPC in ein kleines Meßlabor.

Beginnen wir mit der Beschreibung der einzelnen Baugruppen und Bauteile.

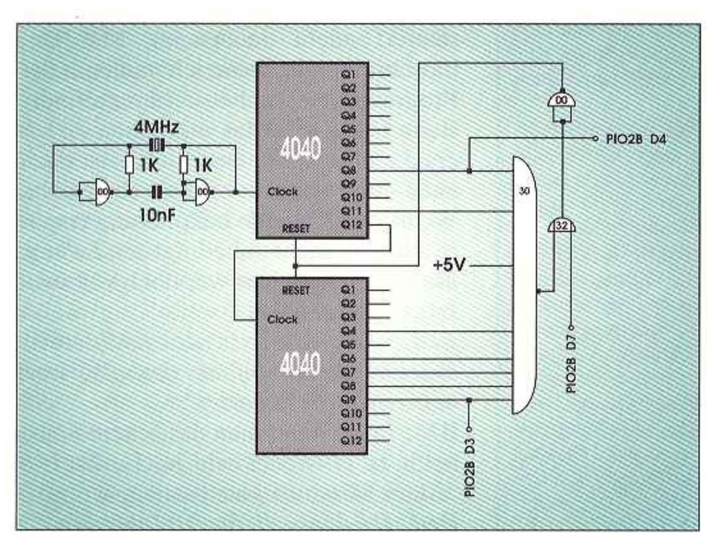
Das wichtigste einer Computererweiterung ist unbestritten die Einheit, die für die Kommunikation mit dem Rechner zuständig ist. Sie wird durch den Adreßdecoder gebildet. In diesem Fall sorgt der Decoder dafür, daß sich die Schaltung nur in einem Adreßbereich von &F9F0 bis &F9F8 angesprochen fühlt. Diese Baugruppe besteht aus drei ICs. Der Adreßdekoder wird durch die ICs 1, 2 und 3 gebildet.

Der Schaltkreistester im Detail:

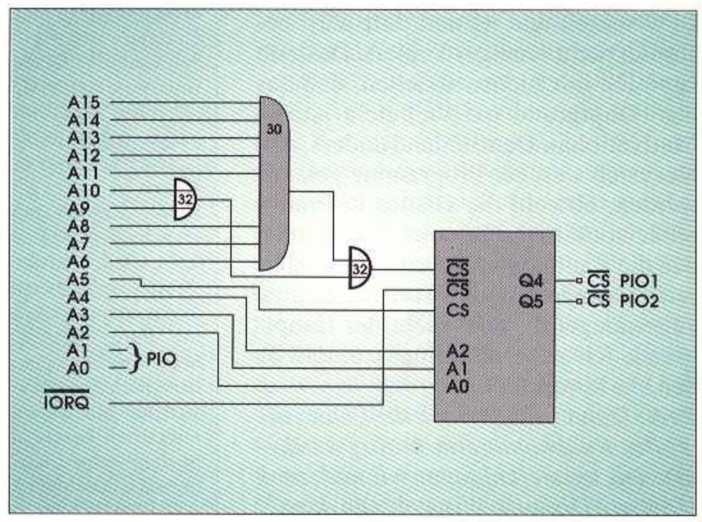
Ein Schaltkreis, der das Tor zwischen Rechner und Peripherie bildet, ist die PIO. Die Eigenschaften dieses Schaltkreises werden durch bestimmte Befehle festgelegt. So kann für jedes einzelne Bit der Peripherieseite bestimmt werden, ob es Ein- oder Ausgang ist oder nicht benutzt wird. Diese Eigenschaften sind beispielsweise bei der Realisierung des IC-Testers besonders wichtig.

Leider sind aber die Anschlüsse einer solchen PIO nicht besonders belastbar. So reicht der Ausgangsstrom eines Ausgangs nicht aus, um einen TTL-Schaltkreis mit dem nötigen Strom zu versorgen. Zur Verstärkung werden deshalb hier Optokoppler eingesetzt. So kann also durch die Ausgabe eines High-Signals auf der gewünschten Leitung der Schaltkreis auf dem entsprechenden Pin mit der Betriebsspannung versorgt werden.

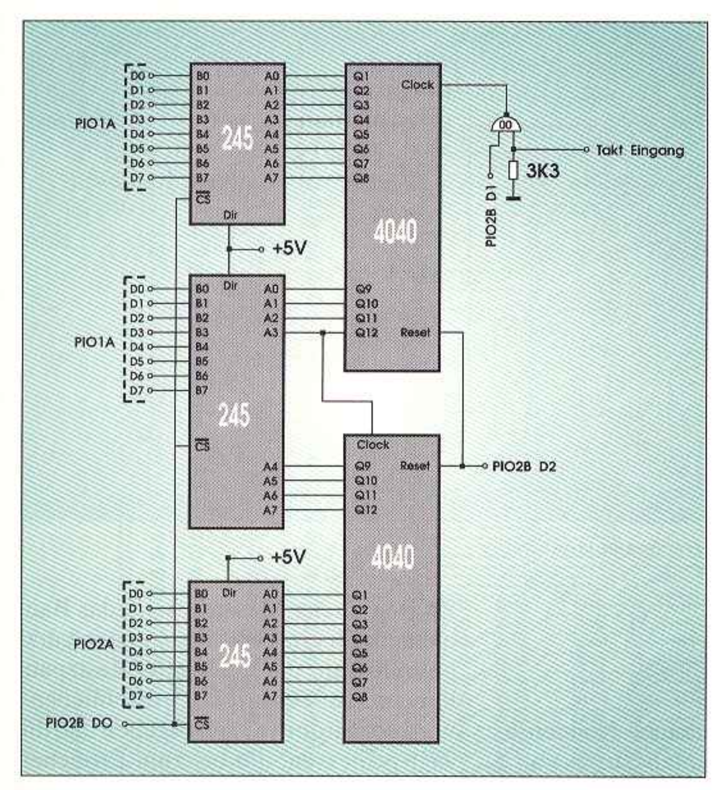
Ebenso wird bei der Masseversorgung vorgegangen. Zusätzlich werden, um die Schaltung nicht unnötig zu komplizieren, einige Kombinationen durch Jumper eingestellt.



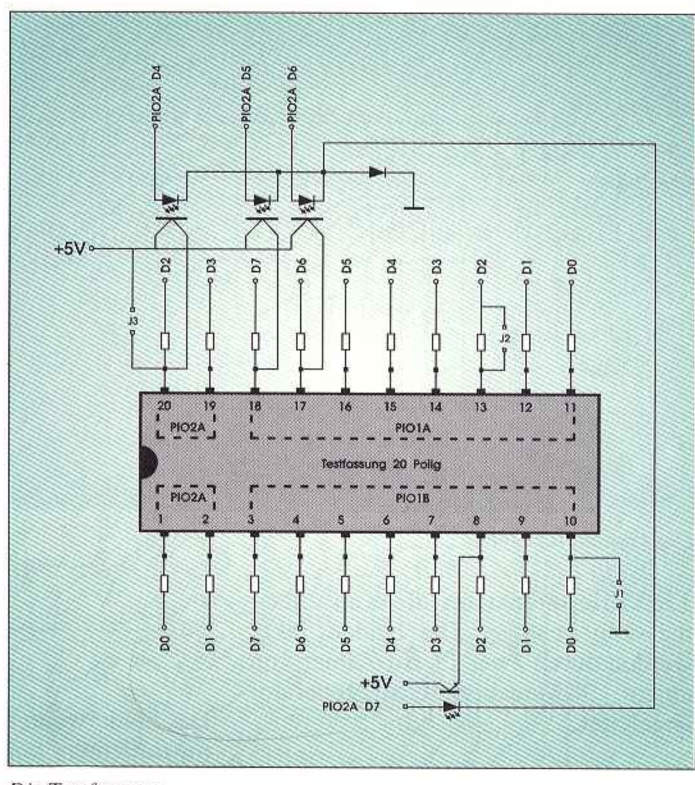
Der Frequenzgeber



Der Adreß-Dekoder



Der Frequenzzähler



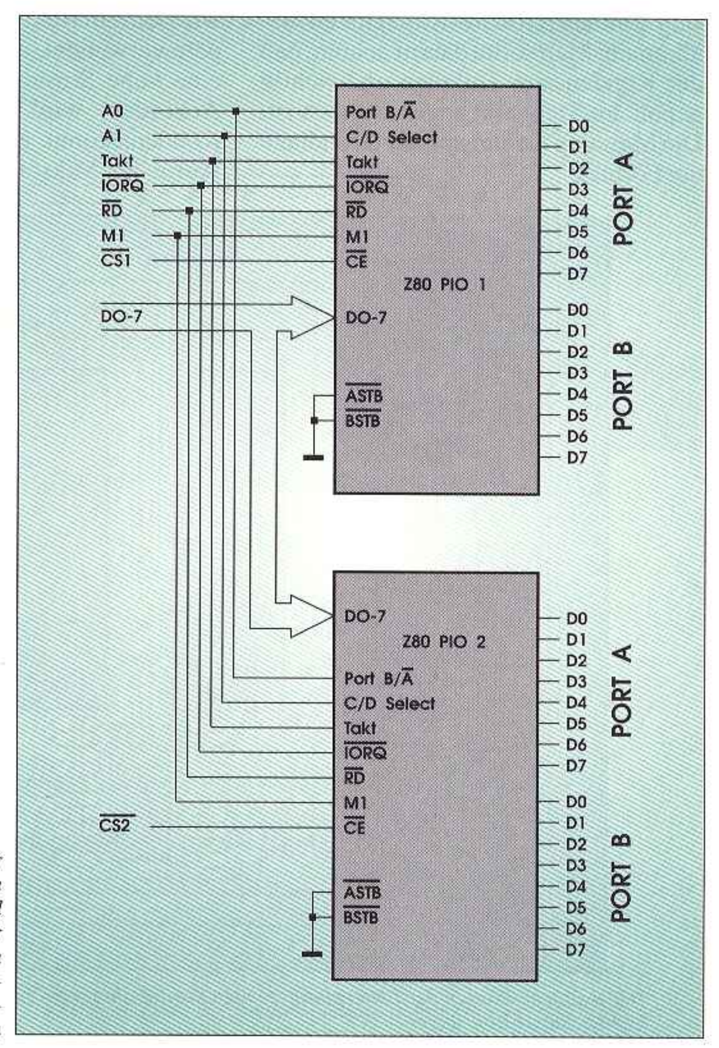
Die Testfassung

Der Frequenzzähler:

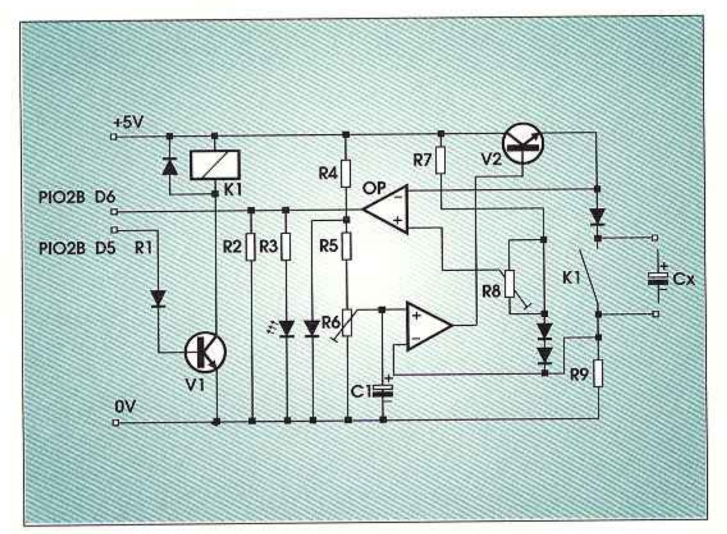
Zum genauen Abgleich von Schaltungen mit eigenem Frequenzgenerator ist unbedingt ein Zähler erforderlich. Ein solches Gerät mit dem benötigten Frequenzumfang ist nicht gerade billig. Deshalb wurde es ebenfalls in unserem Meßlabor mit vorgesehen. Einzige Voraussetzung, die die zu messende Frequenz erfüllen muß: Sie muß TTL-Pegel haben und darf nicht größer als 16 Megahertz sein.
Die Eingangsfrequenz wird auf zwei 12-Bit-Zähler geleitet. Wenn man diese eine halbe Sekunde lang zählen läßt, kann man an ihren Ausgängen die halbe Frequenz in Hertz ablesen. Dies geschieht ebenfalls über eine PIO. Aus CPC-technischen Gründen kann es bei Messung höherer Frequenzen bis zu einem Fehler von 1000 Hertz kommen. Das wird durch die interruptgesteuerte Arbeit und durch Speicherzugriffe des Gate-Arrays, wobei durch eine Logik für diese Zeit Wait-Zyklen erzeugt werden, verursacht. In beiden Fällen kann die Meßzeit nicht eingehalten werden.

Der Frequenzgeber:

Der Frequenzgeber stellt die beiden für die Messung benötigten Takte (2 Hz, 15625 Hz) bereit. Der 2-Hertz-Takt wird für den Frequenzzähler und der



Die PIOs sorgen für die Kommunikation zwischen dem CPC und der Schaltung. Über die vier Ports werden Daten bit- oder byteweise ein- und ausgegeben



Das Kapazitätsmeßgerät

Der Spannungswandler

15625-Hertz-Takt für die Kapazitätsmessung benötigt. Die beiden Takte werden durch zwei 12-Bit-Zähler von einem 4-Megahertz-Quarzgenerator abgeleitet.

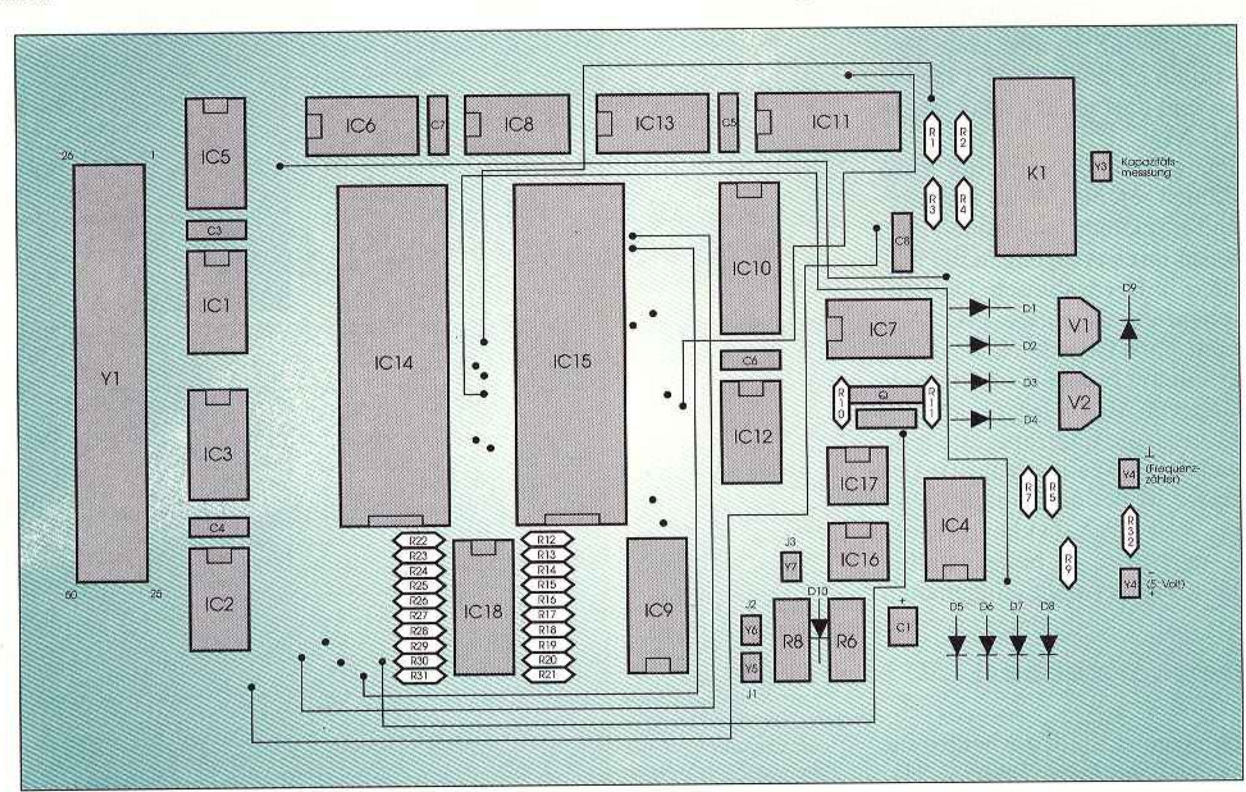
Das Kapazitäts-Meßgerät:

Mit dem Kapazitätsmeßgerät lassen sich einfach größere Kapazitäten bestimmen. Man sollte darauf achten, daß der Kondensator entladen und richtig gepolt am Meßgerät angeschlossen ist. Die Messung erfolgt auf der Grundlage der Formel:

$$F = \frac{V \quad * \quad s}{A}$$

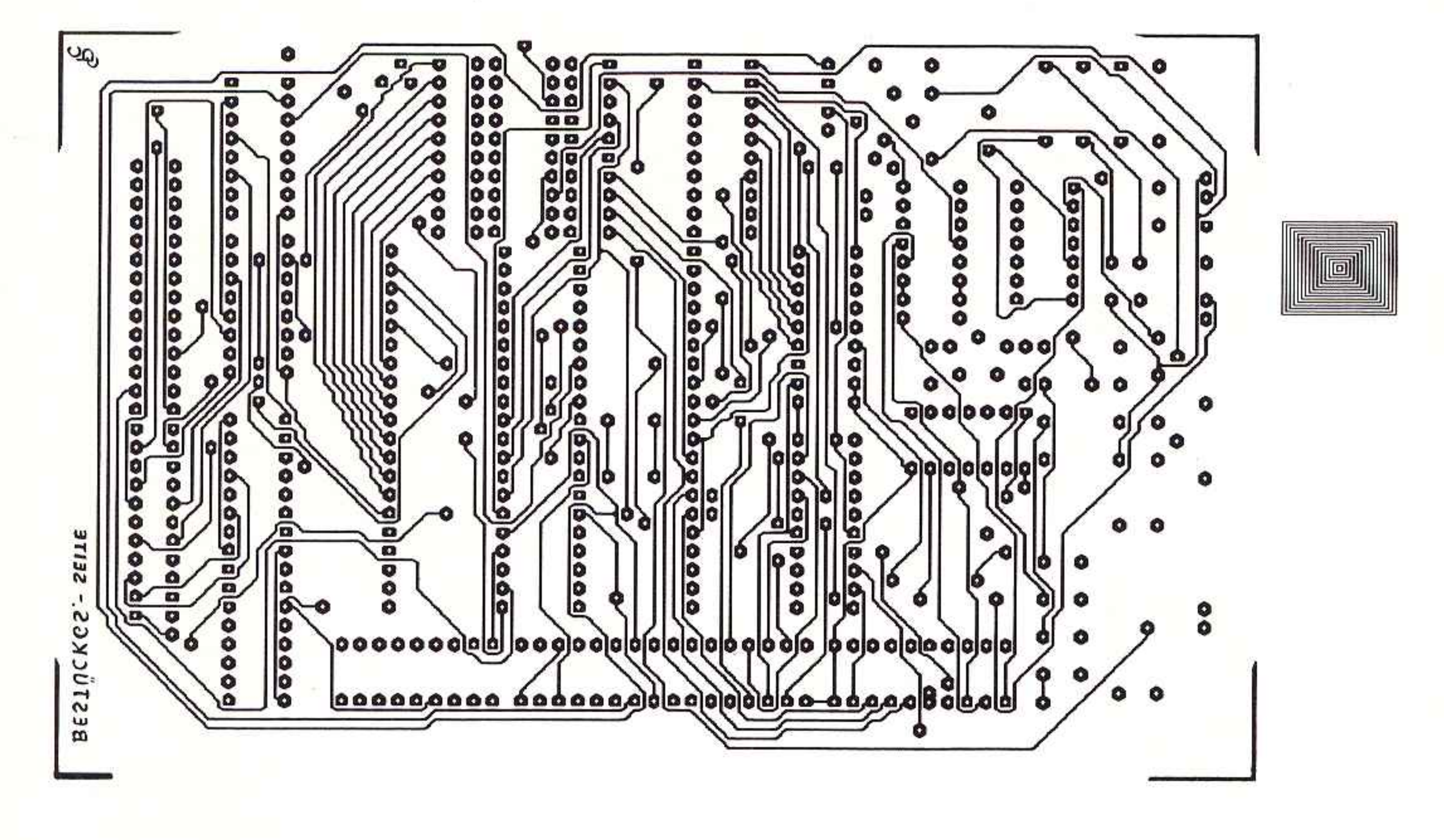
Das heißt, wenn man einen Kondensator mit einem konstanten Strom von 1 Ampere eine Sekunde lang lädt und an seinen Polklemmen dann eine Spannung von 1 Volt anliegt, so beträgt seine Kapazität 1 Farad. In der Schaltung wird lediglich ein Strom von 1 Milliampere verwendet, um die Verluste durch eine hohe Stromstärke zu verringern. Außerdem wird es damit einfacher, Kondensatoren zu bestimmen, deren Werte im Mikrofarad-Bereich liegen.

Zu Beginn der Messung wird das Bauelement für einige Millisekunden kurzgeschlossen, um eventuelle Restspannungen zu beseitigen. Anschließend wird der Stromfluß freigegeben und die Zeit gemessen, die bis zum Erreichen der 1-Volt-Grenze vergeht. Die Zeitmessung geschieht durch Zählung des 15625-Hertz-Takts. Der gezählte Wert wird mit 0.065 multipliziert, und man erhält als Ergebnis die Kapazität des Kondensators in Mikrofarad. Die Messung wird dreimal wiederholt, um ein möglichst reales Ergebnis zu erhalten.
Die Genauigkeit der Messung hängt stark von der sorgfältigen Einstellung



Der Bestückungsplan der Leiterplatte. Achten Sie auf die sorgfältige Herstellung der Durchkontaktierungen und Einlöten von Drähten in nicht benutzte Bohrungen

Das Leiterplattenlayout der Leiterseite (unten). Den hiervon erstellten Film legen Sie so auf die Leiterplatte, daß die Schrift richtig darauf erscheint



Das Layout der Bestückungsseite. Achten Sie hier ebenfalls darauf, daß die Beschriftung auf der Leiterplatte richtigherum erscheint

des Konstantstroms von 1 Milliampere und des Wertes der Schwellspannung von 1 Volt ab. Zur Einstellung des Stroms wird ein Amperemeter verwendet, das statt eines zu messenden Kondensators angeschlossen wird. Mit dem Widerstand R6 wird nun der genaue Wert 1 Milliampere eingestellt.
Die Einstellung der Schwellspannung ist etwas komplizierter. Hierzu benötigen Sie ein Voltmeter und ein Spindelpotentiometer mit dem Wert 2 Kiloohm. Das Spindelpotentiometer schließt man ebenfalls an den Meßanschluß und parallel dazu das Voltmeter an. Am Spindeltrimmer stellt man nun genau eine Spannung von 1 Volt ein. Nun dreht man an R8 so lange, bis die gelbe Leuchtdiode flackert. Den gesamten Justiervorgang sollte man möglichst schnell vornehmen, da sonst Fehler durch Erwärmung der Bauelemente auftreten können.

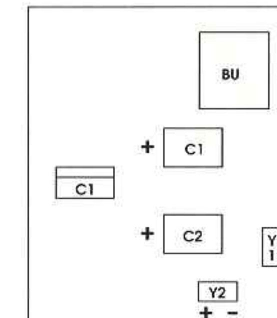
Das Netzteil:

Das Netzteil ist mit einem Spannungsregler 7805 aufgebaut. Dieser bildet aus den 12 Volt vom CPC-Monitor eine konstante Spannung von 5 Volt. Zur ausreichenden Kühlung ist der 7805 auf einem Kühlblech montiert. Beim Aufbau sollte man vor allem auf die richtige Polung (12-Volt-Stecker vom Monitor: außen plus, innen minus) achten.
Das ganze Gerät wird auf einer doppelseitigen Platine im Euroformat aufgebaut. Bevor Sie mit dem Bestücken beginnen, untersuchen Sie die Leiterbahnen genauestens auf Fehler wie Unterbrechungen oder Kurzschlüsse. Wichtig ist das vor allem auf der Bestückungsseite. Befindet sich ein Fehler unter einem Bauelement, ist er im nachhinein sehr schwer zu lokalisieren und vor allem zu beseitigen. Achten Sie auch darauf, die Bauelemente auf der richtigen Seite zu bestücken. Vor dem Bestücken sollten Sie unbedingt alle Bohrungen, die nicht durch Bauelemente benutzt werden, durch Einlöten von Drähtchen in Durchkontaktierungen verwandeln. Und vergessen Sie auch die Brücken nicht; sie sind im Bestückungsplan eingezeichnet. Haben Sie nun alles komplett aufgebaut und eingehend geprüft, geht es an den ersten Test. Verbinden Sie alle Leitungen ordnungsgemäß, und schalten Sie den CPC ein. Erscheint nun kein Bild, oder zeigt der CPC irgendein seltsames Verhalten, schalten Sie ihn unverzüglich aus. Nun heißt es leider Fehler suchen. Läuft der CPC, kontrollieren Sie die einzelnen Teile. Zeigt die Schaltung keinerlei Reaktion, beginnen Sie zuerst im Bereich des Adreßdecoders, nach einem Fehler zu suchen. Funktioniert dagegen nur ein Teilbereich, begibt man sich von den PIOs ausgehend auf die Fehlersuche. Auch eine Kontrolle der Betriebsspannung und der Stromaufnahme kann einen Hinweis auf Fehler geben. Die Stromaufnahme sollte bei abgefallenem Relais 0,35 Ampere nicht überschreiten. Ist der Wert zu hoch, weist dies auf einen Kurzschluß oder ein defektes Bauelement hin.

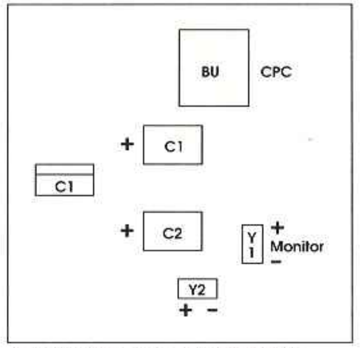
Die Software:

Die Software besteht aus zwei Teilen, dem Meßprogramm und dem Programm zum Erweitern der Bauelementeliste des IC-Testers. Beide Prgramme sind in BASIC geschrieben, besitzen aber einen in Maschine geschriebenen Anteil. Vor dem Start des Meßprogramms TESTP.BAS sollte die Schaltung angeschlossen und eingeschaltet sein. Es darf kein Meßobjekt angeschlossen sein, sonst wird eine Fehlermeldung ausgegeben. Nach dem korrekten Anlauf des Programms sehen Sie ein Menü, in dem Sie unter den drei Meßmöglichkeiten wählen können. Da das Programm Ihnen die nötigen Bedienungshinweise gibt, ist es leicht zu bedienen.

Bei der Schaltkreistestung ist zu beachten, daß der Testschaltkreis erst nach dem richtigen Setzen der Jumper aufgesteckt wird. Die Position der Jumper teilt Ihnen das Programm mit. Zum Schaltkreistestprogramm gehört die Datei TESTICDA.LDR. In ihr sind die Daten der Schaltkreise abgelegt.
Die Datendatei kann mit dem Programm TESTS.BAS vervollständigt werden. Es können ICs mit 14, 16 und 20 Polen getestet werden. Die Schaltkreise müssen stets bündig mit der unteren Kante auf die Fassung gesteckt werden. Nach dem Start zeigt Ihnen das Programm die Anzahl der bisher gespeicherten Schaltkreismasken an. Eine Maske ist das Modell eines Schalt-



Layout des Spannungswandlers



Bestückungsplan des Wandlers

Bauteileliste:

Schaltkreise:		
2	Z80 PIO	(IC14, IC15)
2	74LS30	(IC1, IC8)
1	74LS32	(IC2)
1	74LS138	(IC3)
3	74LS245	(IC9, IC10, IC11)
4	4040	(IC5, IC6, IC12, IC13)
1	74LS00	(IC7)
Widerstände:		
3	3,3 kΩ	(R4, R7, R32)
1	680 Ω	(R1)
1	2,7kΩ	(R2)
1	3,9 kΩ	(R5)
1	100 Ω	(R9)
2	1 kΩ	(R10, R11)
2	2 kΩ Spindel-Poti	(R6, R8)
20	220 Ω	(R12–R31)
Kondensatoren:		
1	0,47 µF	(C1)
6	100 nF	(C3–C8)
1	10 nF	(C2)
1	220 µF	(C9)
1	10 µF	(C10)
Sonstiges:		
1	7805	
2	ILD74 (CNY 74-2)	(IC16, IC17)
2	BC 546	(V1, V2)
1	LM324	(IC4)
9	1N4148	(D1, D2, D4–D10)
1	LED gelb	(D3)
1	Kleinrelais	(K1)
1	Quarz 4 Mhz	(Q)
1	Kleinspannungsstecker	
1	Kleinspannungsbuchse	
1	Jumper	(Y5–Y7)
1	Stecker 50polig (Centronics (6128) oder Leiterplattenstecker (464, 664)	(Y1)
1	50poliges Flachbandkabel	
3	2polige Stiftleisten	(Y2–Y4)

kreises, in dem die Ein- und Ausgänge des IC festgelegt sind. Bei einem Schaltkreis wie dem 74245 können die Anschlüsse Ein- oder Ausgang sein. In einem solchen Fall müssen dann zwei Masken oder möglicherweise mehr angelegt werden.
Zur Eingabe wird auf dem Bildschirm ein stilisierter Schaltkreis abgebildet. Die Pins kennzeichnen Sie nun durch Angabe von E oder A, ob sie Ein- oder Ausgang sind. Die Pins zur Stromversorgung (5 Volt und Masse) werden als Eingänge gekennzeichnet. Rechts oben im Bild wird das gerade bezeichnete Pin angezeigt. Mit den Cursortasten wechseln Sie zu den anderen Anschlüssen. Werfen Sie nun einen Blick auf den Schaltplan, hier können Sie erkennen wie die Jumper zu stecken sind. Befindet sich die Stromversorgung an den Pins 5 und 10 des Schaltkreises, so stecken Sie Jumper 2. Befindet sie sich dagegen an den Anschlüssen 10 (8 oder 7) und 20 (16, 14), so stecken Sie Jumper 1 und 2. Nun werden Sie zum Aufstecken eines intakten Testbauelements und zur Angabe der Pins zur Stromversorgung aufgefordert. Anschließend geben Sie für die entsprechenden Testdurchläufe noch die Vorgabe von 0 oder 1 an den Pins an. Das Programm stellt nun die Pegel an den Ausgängen fest und speichert diese.
Wir hoffen, daß Sie nun Lust auf einen Nachbau bekommen haben, und wünschen viel Spaß bei der Arbeit mit der Meßerweiterung.

Markus Müller/jg

```
100 'TESTP.BAS                                 [114]
110 'CPC 6128, 664                             [953]
120 '(c)1992 Markus Mueller                    [806]
130 '   & CPC International                    [2118]
140 '                                          [117]
150 MEMORY &3FFF:LOAD"testicda.bin",&4000:    [6378]
LOAD"testb.bin",&A000:CALL &A000
160 MODE 1:INK 0,1:INK 1,24:INK 2,12          [2281]
170 WINDOW #1,9,31,9,21:PAPER #1,2:CLS #1     [1390]
180 WINDOW 10,30,10,20:PAPER 0:CLS            [1721]
190 WINDOW #3,10,31,24,24:PRINT#3,"CPC  Zu    [4668]
satzmessgeraet"
200 WINDOW #2,2,39,2,3:CLS #2                 [1251]
210 PRINT #2,CHR$(24);" Capazitaet ";CHR$(    [7597]
24);" - ";CHR$(24);" Frequenz ";CHR$(24);"
 - ";CHR$(24);" IC-Test ";CHR$(24)
220 IF PEEK(&A069)>0 THEN LOCATE #1,9,13:P    [8311]
RINT#1,"Fehler" ELSE LOCATE #1,4,13:PRINT#
1,"System in Ordnung"
230 pf$=CHR$(&F4):pf=1                        [1055]
240 IF INKEY(8)>-1 AND pf>1 THEN pf=pf-1      [1259]
250 IF INKEY(1)>-1 AND pf<3 THEN pf=pf+1      [711]
260 IF INKEY(18)>-1 THEN 310                  [844]
270 IF pf=1 THEN LOCATE #2,1,2:PRINT #2,SP    [3019]
ACE$(5);pf$;SPACE$(15);
280 IF pf=2 THEN LOCATE #2,1,2:PRINT #2,SP    [3048]
ACE$(19);pf$;SPACE$(15);
290 IF pf=3 THEN LOCATE #2,1,2:PRINT #2,SP    [2937]
ACE$(32);pf$;
300 GOTO 240                                  [421]
310 CLEAR INPUT:ON pf GOTO 320,460,540        [1296]
320 CLS:LOCATE 4,1:PRINT"Messbereich von":    [5624]
LOCATE 4,2:PRINT"0.07 uF bis xx."
330 LOCATE 8,4:PRINT"<Taste>";:CALL &BB18     [1520]
340 FOR m=6 TO 8:CALL &A06A                   [1088]
350 d1=PEEK(&A10C):d2=PEEK(&A10D):d3=PEEK(    [3429]
&A10E):d4=PEEK(&A10F)
360 d=d1+d2*&100+d3*&100*&100/2+d4*&100*&1    [3362]
00*&100:c=d*0.065
370 IF c<24 THEN c=ROUND(c*100)/100:GOTO 4    [1896]
10
380 IF c<140 THEN c=ROUND(c*10)/10:GOTO 41    [3085]
0
390 IF c<700 THEN c=ROUND(c):GOTO 410         [2068]
400 c=ROUND(c/10)*10                          [1911]
410 LOCATE 1,m:PRINT m-5;"Messung";c;:LOCA    [3871]
TE 18,m:PRINT "uF";
420 NEXT                                      [350]
430 LOCATE 2,10:PRINT"Neue Messung (J/N)"     [2811]
440 a$=LOWER$(INKEY$):IF a$="" THEN 440       [1494]
450 IF a$="j" THEN 320 ELSE 240               [1784]
460 CLS:CALL &A110:PRINT"    Frequenzmesse    [4305]
r":PRINT"        0-16 MHz"
470 LOCATE 8,4:PRINT"<Taste>":CALL &BB18      [2164]
480 CALL &A126:f1=PEEK(&A16F):f2=PEEK(&A17    [2411]
0):f3=PEEK(&A171)
490 frequ=(f3+f2*&100+f1*&100*&100)*2         [1966]
500 LOCATE 6,7:PRINT frequ;"Herz"             [3642]
510 LOCATE 2,10:PRINT"Neue Messung (J/N)"     [2811]
520 a$=LOWER$(INKEY$):IF a$="" THEN 520       [1526]
530 IF a$="j" THEN 460 ELSE 240               [1947]
540 CLS:PRINT"      IC - Tester"              [1951]
550 PRINT" ";PEEK(&4000);"bekannte IC's"      [3302]
560 PRINT:INPUT" Welchen Typ? 74",typ         [1629]
570 IF typ=90 OR typ=93 THEN PRINT:PRINT"     [4860]
 Jumper 2 stecken":GOTO 600
580 IF typ=245 THEN PRINT:PRINT" Jumper 1+    [2710]
3 stecken":GOTO 600
590 PRINT:PRINT"  Jumper 1 stecken"           [1971]
600 LOCATE 8,7:PRINT"<Taste>"                 [2384]
610 CALL &BB18:LOCATE 8,7:PRINT"         "    [2239]
620 adr=&4000:st=PEEK(adr)                    [1126]
630 FOR z=1 TO st:adr=adr+1:a1=PEEK(adr):a    [3020]
1=a1*&100
640 adr=adr+1:a2=PEEK(adr):a=a1+a2            [1726]
650 adr=adr+1:IF a<>typ THEN f=PEEK(adr):a    [5724]
dr=adr+f:NEXT ELSE 660
660 adr=adr+1:masken=PEEK(adr)                [1211]
670 FOR mk=1 TO masken                        [418]
680 adr=adr+1:OUT &F9F2,&3:OUT &F9F2,&CF:O    [4121]
UT &F9F2,PEEK(adr)
690 adr=adr+1:OUT &F9F3,&3:OUT &F9F3,&CF:O    [3776]
UT &F9F3,PEEK(adr)
700 adr=adr+1:OUT &F9F6,&3:OUT &F9F6,&CF:O    [3933]
UT &F9F6,PEEK(adr)
710 adr=adr+1:funk=PEEK(adr)                  [1583]
720 FOR fk=1 TO funk                          [580]
730 adr=adr+1:OUT &F9F0,PEEK(adr)             [1976]
740 adr=adr+1:OUT &F9F1,PEEK(adr)             [2103]
750 adr=adr+1:OUT &F9F4,PEEK(adr)             [998]
760 adr=adr+1:IF INP(&F9F0)<>PEEK(adr) THE    [2327]
N 820
770 adr=adr+1:IF INP(&F9F1)<>PEEK(adr) THE    [2307]
N 820
780 adr=adr+1:IF INP(&F9F4)<>PEEK(adr) THE    [3024]
N 820
790 NEXT:NEXT                                 [1022]
800 IF z=st+1 THEN PRINT CHR$(24);"IC nich    [4309]
t gespeichert ";CHR$(24):GOTO 830
810 LOCATE 5,8:PRINT "IC in Ordnug":GOTO 8    [1686]
30
820 LOCATE 5,8:PRINT CHR$(24);" IC deffekt    [2399]
 ";CHR$(24)
830 OUT &F9F2,&CF:OUT &F9F2,&FF               [805]
840 OUT &F9F3,&CF:OUT &F9F3,&FF               [2055]
850 OUT &F9F6,&CF:OUT &F9F6,&FF               [1467]
860 LOCATE 2,10:PRINT"Noch ein Test (J/N)"    [2502]
870 a$=LOWER$(INKEY$):IF a$="" THEN 870       [1504]
880 IF a$="j" THEN 540 ELSE 240               [1947]
```

```
100 'TESTICDA.LDR                             [401]
110 'erzeugt TESTICDA.BIN                     [1750]
120 '(c)1992 Markus Mueller                   [806]
130 '   & CPC International                   [2118]
140 DATA 14,00,00,1e,01,89,92,00,04,40,00,    [1849]
40
150 DATA c9,92,40,64,48,40,ed,da,40,52,24     [1865]
160 DATA 40,db,b6,40,76,6c,40,f6,ec,40,00     [1608]
170 DATA 00,02,1e,01,24,48,00,04,40,00,40     [1660]
180 DATA 64,48,40,52,24,40,52,24,40,49,12     [1516]
190 DATA 40,49,12,40,5b,36,40,5b,36,40,00     [1920]
200 DATA 00,04,12,01,95,aa,00,02,40,00,40     [1733]
210 DATA d5,aa,40,6a,54,40,ea,d4,40,00,00     [2057]
220 DATA 07,12,01,95,aa,00,02,40,00,40,c0     [2303]
230 DATA 80,40,6a,54,40,ff,fe,40,00,00,08,    [1540]
1e,01,89
240 DATA 92,00,04,40,00,40,c0,80,40,64,48     [1898]
250 DATA 40,e4,c8,40,52,24,40,d2,a4,40,76     [1717]
260 DATA 6c,40,ff,fe,40,00,00,0a,1e,01,11     [1772]
270 DATA 02,00,04,62,10,40,73,12,40,44,48     [1750]
280 DATA 40,55,4a,40,48,24,40,59,26,40,6e     [1404]
290 DATA 7c,40,6e,7c,40,00,00,14,18,01,01     [2167]
300 DATA 02,00,03,52,48,40,53,4a,40,64,24     [1683]
310 DATA 40,65,26,40,76,6c,40,76,6c,40,00     [2333]
320 DATA 00,1b,1e,01,11,02,00,04,62,10,40     [1273]
330 DATA 62,10,40,44,48,40,44,48,40,48,24     [1237]
```

```
340 DATA 40,48,24,40,40,00,40,51,02,40,00  [1067]
350 DATA 00,1e,18,01,01,00,00,03,48,54,40  [2027]
360 DATA 49,54,40,50,2a,40,51,2a,40,58,7e  [1612]
370 DATA 40,58,7e,40,00,00,20,18,01,09,12  [1416]
380 DATA 00,03,40,00,40,40,00,40,64,48,40  [1694]
390 DATA 6d,5a,40,52,24,40,5b,36,40,00,00  [2243]
400 DATA 2a,42,01,07,fe,00,0a,80,00,20,87  [1105]
410 DATA 7e,20,c0,00,20,c7,be,20,a0,00,20  [1933]
420 DATA a7,de,20,e0,00,20,e7,ee,20,90,00  [1405]
430 DATA 20,97,f6,20,d0,00,20,d7,fa,20,b0  [2896]
440 DATA 00,20,b7,fc,20,f0,00,20,f6,fe,20  [961]
450 DATA 88,00,20,8d,fe,20,c8,00,20,cb,fe  [1281]
460 DATA 20,00,00,2f,48,01,7f,10,00,0b,80  [1714]
470 DATA 28,20,a0,38,20,80,2a,20,f3,3a,20  [1609]
480 DATA 80,a8,20,c4,b8,20,80,aa,20,c1,ba  [1704]
490 DATA 20,80,68,20,93,78,20,80,6a,20,89  [1593]
500 DATA 7a,20,80,e8,20,98,f8,20,80,ea,20  [2191]
510 DATA e3,fa,20,80,2c,20,80,3c,20,80,2e  [2165]
520 DATA 20,83,3e,20,80,08,20,80,18,20,00  [1782]
530 DATA 00,30,48,01,7f,10,00,0b,80,28,20  [1339]
540 DATA df,38,20,80,2a,20,8c,3a,20,80,a8  [1228]
550 DATA 20,bb,b8,20,80,aa,20,be,ba,20,80  [2519]
560 DATA 68,20,ec,78,20,80,6a,20,f6,7a,20  [1080]
570 DATA 80,e8,20,e7,f8,20,80,ea,20,9c,fa, [1614]
20
580 DATA 80,2c,20,ff,3c,20,86,2e,20,fc,3e  [1682]
590 DATA 20,80,08,20,ff,18,20,00,00,4a,24  [1143]
600 DATA 01,03,06,00,05,44,08,40,45,0a,40  [1580]
610 DATA 60,40,40,62,44,40,64,48,40,66,4c  [1664]
620 DATA 40,6c,58,40,6d,5a,40,5c,38,40,5d  [2284]
630 DATA 3a,40,00,00,56,1e,01,09,12,00,04  [1718]
640 DATA 40,00,40,40,00,40,64,48,40,6d,5a  [1208]
650 DATA 40,52,24,40,5b,36,40,76,6c,40,76  [1085]
660 DATA 6c,40,00,00,5a,4d,01,1b,00,00,0c  [1397]
670 DATA 00,34,80,00,34,80,40,44,80,40,44  [1355]
680 DATA 80,00,04,80,12,04,80,40,44,80,52, [2510]
44,80,00,04,80,01
690 DATA 04,80,40,44,80,41,44,80,00,04,80  [1968]
700 DATA 13,04,80,40,44,80,53,44,80,00,04  [2173]
710 DATA 80,08,04,80,40,44,80,48,44,80,00  [1781]
720 DATA 04,80,10,04,80,00,07,80,18,07,80  [1939]
730 DATA 00,5d,6b,01,1b,00,00,11,00,34,80  [1554]
740 DATA 00,34,80,40,44,80,40,44,80,00,04  [1861]
750 DATA 80,12,04,80,40,44,80,52,44,80,00  [1567]
760 DATA 04,80,01,04,80,40,44,80,41,44,80  [1771]
770 DATA 00,04,80,13,04,80,40,44,80,53,44  [1531]
780 DATA 80,00,04,80,08,04,80,40,44,80,48  [1644]
790 DATA 44,80,00,04,80,1a,04,80,40,44,80  [1529]
800 DATA 5a,44,80,00,04,80,09,04,80,40,44  [1739]
810 DATA 80,49,44,80,00,04,80,1b,04,80,40, [1806]
44
820 DATA 80,5b,44,80,00,04,80,00,04,80,00, [1763]
8a
830 DATA 3b,01,7f,02,00,09,80,00,20,ff,02  [1758]
840 DATA 20,80,04,20,bf,06,20,80,84,20,df  [2325]
850 DATA 86,20,80,44,20,ef,46,20,80,c4,20  [1457]
860 DATA f7,c6,20,80,24,20,fb,26,20,80,a4  [3283]
870 DATA 20,fd,a6,20,80,64,20,fe,66,20,80  [1745]
880 DATA e4,20,ff,e4,20,00,9d,23,01,09,12  [1954]
890 DATA 00,05,f6,6c,20,f6,6c,20,92,24,20  [1022]
900 DATA 92,24,20,a4,48,20,ad,5a,20,a4,c8  [1808]
910 DATA 20,a4,c8,20,92,a4,20,9b,b6,20,00  [1709]
920 DATA f5,2d,02,ff,00,00,03,00,aa,1d,ff  [1446]
930 DATA aa,1d,00,aa,15,55,aa,15,00,54,17  [1987]
940 DATA aa,54,17,00,fe,02,03,55,00,1c,55  [1691]
950 DATA fe,1e,55,00,14,55,aa,14,aa,00,14  [1648]
960 DATA aa,54,16                          [489]
970 MEMORY &3FFF:adr=&4000:su=0            [1400]
980 FOR ad=adr TO adr+917:READ d$:d=VAL("& [5300]
"+d$):su=su+d:POKE ad,d:NEXT
990 IF su<>69045 THEN PRINT "FEHLER"+CHR$( [2487]
7):END
1000 SAVE"testicda.bin",b,&4000,918        [2480]
```

```
100 'TESTB.LDR                             [376]
110 'erzeugt TESTB.BIN                     [1210]
120 '(c)1992 Markus Mueller                [806]
130 '    & CPC International               [2118]
140 DATA 01,F7,F9,3E,03,ED,79,3E           [1111]
150 DATA CF,ED,79,3E,58,ED,79,0E           [1002]
160 DATA F5,3E,85,ED,79,0E,F2,3E           [1476]
170 DATA 4F,ED,79,0E,F3,ED,79,0E           [1500]
180 DATA F6,ED,79,0E,F5,ED,78,CB           [1636]
190 DATA 77,C2,61,A0,3E,A5,ED,79           [1953]
200 DATA 3E,03,32,67,A0,3E,FF,32           [1568]
210 DATA 68,A0,3A,68,A0,D6,01,32           [1402]
220 DATA 68,A0,C2,3A,A0,3A,67,A0           [1037]
230 DATA D6,01,32,67,A0,C2,35,A0           [949]
240 DATA ED,78,CB,77,CA,61,A0,3E           [878]
250 DATA 85,ED,79,3E,00,32,69,A0           [947]
260 DATA C9,3E,01,32,69,A0,C9,00           [1624]
270 DATA 00,00,3E,A5,21,00,00,22           [992]
280 DATA 0E,A1,22,0C,A1,01,F5,F9           [862]
290 DATA 11,00,00,ED,79,3E,C8,CD           [2002]
300 DATA ED,A0,3E,85,ED,79,3E,F5           [1185]
310 DATA 32,67,A0,3A,67,A0,D6,01           [1308]
320 DATA 32,67,A0,C2,8B,A0,3E,02           [994]
330 DATA 32,67,A0,3A,67,A0,D6,01           [1308]
340 DATA 32,67,A0,C2,9B,A0,ED,78           [1476]
350 DATA CB,77,CA,DF,A0,CB,67,C2           [989]
360 DATA A6,A0,ED,78,CB,77,CA,DF           [496]
370 DATA A0,CB,67,CA,B2,A0,ED,5B           [946]
380 DATA 0C,A1,21,01,00,19,22,0C           [926]
390 DATA A1,11,FF,FF,ED,52,C2,A6           [925]
400 DATA A0,ED,5B,0E,A1,21,01,00           [441]
410 DATA 19,22,0E,A1,C3,A6,A0,3E           [359]
420 DATA A5,ED,79,3E,C8,CD,ED,A0           [987]
430 DATA 3E,85,ED,79,C9,32,67,A0           [1672]
440 DATA 3E,FF,32,68,A0,3A,68,A0           [1425]
450 DATA D6,01,32,68,A0,C2,F5,A0           [1041]
460 DATA 3A,67,A0,D6,01,32,67,A0           [677]
470 DATA C2,F0,A0,C9,02,00,00,00           [707]
480 DATA 01,F2,F9,3E,4F,ED,79,0E           [1144]
490 DATA F3,ED,79,0E,F6,ED,79,0E           [1243]
500 DATA F5,3E,00,ED,79,C9,F3,01           [1673]
510 DATA F5,F9,ED,78,CB,5F,C2,2A           [1390]
520 DATA A1,ED,78,CB,5F,CA,31,A1           [1532]
530 DATA 3E,02,ED,79,ED,78,CB,5F           [1881]
540 DATA C2,3C,A1,ED,78,CB,5F,CA           [839]
550 DATA 43,A1,3E,00,ED,79,01,F0           [1438]
560 DATA F9,ED,78,32,70,A1,01,F1           [1082]
570 DATA F9,ED,78,32,71,A1,01,F4           [1345]
580 DATA F9,ED,78,32,6F,A1,01,F5           [1877]
590 DATA F9,3E,85,ED,79,FB,C9,ED           [1358]
600 DATA 00,00,00,00,00,00,00,00           [1005]
610 FOR adr=&A000 TO &A172:READ d$         [1226]
620 POKE adr,VAL("&"+d$):NEXT              [1294]
630 SAVE"testb.bin",b,&A000,&172           [2050]
```

```
100 'TESTS.BAS                             [98]
110 'CPC 6128, 664                         [953]
120 '(c)1992 Markus Mueller                [806]
130 '   & CPC International                [2118]
140 '                                      [117]
150 MODE 2:INK 0,1:INK 1,22:PEN 1:PAPER 0: [6952]
adr%=&2D00:an%=adr%:MEMORY adr%-1:DIM d$(1
0,2),dm(20)
160 FOR a=1 TO 2:FOR b=1 TO 10:da$(b,a)="" [3195]
:NEXT b,a
170 LOCATE 29,2:PRINT"IC-Maskenprogrammier [2964]
ung"
180 LOAD"testicda.bin",adr%:LOAD"tests.bin [5160]
",&A000:CALL &A007
190 st%=PEEK(adr%)                         [329]
200 LOCATE 6,4:PRINT"Vorhandene IC-Masken: [3962]
";st%
210 LOCATE 3,6:INPUT "Welchen IC wollen Si [5884]
e Einprogrammieren(0-65535)";ic:IF ic>6553
5 THEN 210
220 LOCATE 36,4:PRINT"IC-Typ:";ic:LOCATE 3 [5348]
,6:PRINT SPACE$(70)
230 FOR ende%=1 TO st%:adr%=adr%+3:adr%=ad [3182]
r%+PEEK(adr%):NEXT
240 ende%=adr%:adr%=adr%+1:CALL &A002,ic:P [4652]
OKE adr%,PEEK(&A001):POKE adr%+1,PEEK(&A00
0)
250 lgadr%=adr%+2:adr%=adr%+3              [1546]
260 LOCATE 3,6:INPUT"Wieviele Masken benoe [4855]
tigen Sie";mk%
270 POKE adr%,mk%:LOCATE 3,6:PRINT SPACE$( [2507]
70):byt=1
280 FOR m%=1 TO mk%                        [733]
290 LOCATE 52,4:PRINT"Maske:"m%            [2726]
300 PRINT CHR$(23)CHR$(3);:TAG:ORIGIN 65,1 [2972]
56,65,575,135,264
310 MOVE 20,0:RESTORE 1250:FOR d=1 TO 88:R [3735]
EAD x%,y%:DRAWR x%,y%:NEXT
320 DRAWR 420,0:MOVER 0,88:DRAWR -420,0    [3082]
330 MOVE 35,104:FOR z=20 TO 11 STEP -1     [1989]
340 PRINT z;:MOVER 8,0:NEXT                [1729]
350 MOVE 39,-4:FOR z=1 TO 9                [1442]
360 PRINT z;:MOVER 16,0:NEXT               [2108]
370 MOVER -4,0:PRINT 10;:TAGOFF            [1473]
380 LOCATE 26,24:PRINT"Deffinieren Sie die [7247]
";m%;". Maske.":LOCATE 18,25:PRINT"(A=Ausg
ang,E=Eingang, + und - sind Eingaenge)"
390 x=15:y=8:LOCATE x,y:xx=1:yy=1:GOSUB 12 [4255]
```

```
20:yi1=8:yi2=18:yi3=10
400 GOSUB 1150:IF e=1 THEN e=0:GOTO 440      [1431]
410 IF a$="E" THEN da$(xx,yy)=a$:GOSUB 114 [843]
0
420 IF a$="A" THEN da$(xx,yy)=a$:GOSUB 114 [2212]
0
430 GOTO 400                                  [452]
440 FOR a=1 TO 2:FOR b=1 TO 10:IF da$(b,a) [9977]
="" THEN LOCATE 30,23:PRINT"Jeden Pin beze
ichnen":GOTO 400 ELSE NEXT b,a:LOCATE 30,2
3:PRINT SPACE$(20)
450 a=128:c=18:p1a=0:FOR b=3 TO 10:IF da$( [6673]
b,1)="A" THEN p1a=p1a+a:dm(c)=0 ELSE dm(c)
=1
460 a=a/2:c=c-1:NEXT                         [2075]
470 a=128:c=3:p1b=0:FOR b=3 TO 10:IF da$(b [5296]
,2)="A" THEN p1b=p1b+a:dm(c)=0 ELSE dm(c)=
1
480 a=a/2:c=c+1:NEXT                         [2060]
490 a=8:FOR b=1 TO 2:FOR c=2 TO 1 STEP -1: [1694]
IF da$(c,b)="A" THEN p2a=p2a+a
500 a=a/2:NEXT:NEXT:adr%=adr%+1              [1918]
510 IF da$(1,1)="A" THEN dm(20)=0 ELSE dm( [1766]
20)=1
520 IF da$(2,1)="A" THEN dm(19)=0 ELSE dm( [2877]
19)=1
530 IF da$(1,2)="A" THEN dm(1)=0 ELSE dm(1 [1224]
)=1
540 IF da$(2,2)="A" THEN dm(2)=0 ELSE dm(2 [2537]
)=1
550 adr%=adr%+1:POKE adr%,p1a:POKE adr%+1, [4372]
p1b:POKE adr%+2,p2a
560 adr%=adr%+3:LOCATE 20,24:PRINT SPACE$( [5318]
40):byt=byt+3
570 LOCATE 12,25:PRINT"Den Programierenden [4706]
 IC in den Testsockel stecken <Taste>"
580 CLEAR INPUT:CALL &BB18:OUT &F9F2,&CF:O [2638]
UT &F9F2,p1a
590 OUT &F9F3,&CF:OUT &F9F3,p1b:OUT &F9F6, [4243]
&CF:OUT &F9F6,p2a
600 LOCATE 12,25:PRINT SPACE$(60)            [952]
610 LOCATE 20,23:PRINT"Wo ist die Versorgu [5545]
ngsspannung";:INPUT" +";v
620 IF v<>20 AND v<>18 AND v<>17 AND v<>8  [9071]
THEN LOCATE 35,22:PRINT"Fehler";CHR$(7):LO
CATE 54,23:PRINT"    ":GOTO 610
630 LOCATE 35,22:PRINT SPACE$(7)             [1593]
640 IF v=20 THEN LOCATE 15,7:in=&10:im1=0: [3140]
im2=0:im3=4
650 IF v=18 THEN LOCATE 25,7:in=&20:im1=12 [2375]
8:im2=0:im3=0
660 IF v=17 THEN LOCATE 30,7:in=&40:im1=64 [1857]
:im2=0:im3=0
670 IF v=8 THEN LOCATE 50,19:in=&80:im1=0: [2074]
im2=4:im3=0
680 PRINT "+":dm(v)=0                        [1396]
690 LOCATE 54,23:PRINT"    "                 [1716]
700 LOCATE 20,23:PRINT"Wo ist die Versorgu [5496]
ngsspannung";:INPUT" -";v
710 IF v<>13 AND v<>10 THEN LOCATE 35,22:P [5930]
RINT"Fehler";CHR$(7):LOCATE 54,23:PRINT"
  ":GOTO 700
720 LOCATE 35,22:PRINT SPACE$(7)             [1593]
730 IF v=13 THEN LOCATE 50,7                 [1300]
740 IF v=10 THEN LOCATE 60,19                [1216]
750 PRINT "-":dm(v)=0                        [1136]
760 LOCATE 20,23:PRINT SPACE$(60)            [1578]
770 LOCATE 25,23:INPUT "Wieviele Funktiont [3169]
en";f
780 IF f<1 OR f>255 THEN LOCATE 35,22:PRIN [3655]
T"Fehler";CHR$(7):GOTO 770
790 POKE adr%,f:byt=byt+1:adr%=adr%+1        [1663]
800 LOCATE 25,23:PRINT SPACE$(25)            [2030]
810 FOR fd=1 TO f:pa1=0:pa2=0:pb1=0          [2261]
820 LOCATE 20,24:PRINT"Signale eingeben de [3992]
r";fd;". Funktion"
830 po=1                                      [347]
840 IF dm(po)=0 THEN po=po+1:IF po=21 THEN [6723]
 LOCATE 23,23:PRINT"Der IC hat keine Einga
enge":END ELSE 840
850 IF po<11 THEN x=10+po*5:y=19 ELSE x=10 [2149]
+(21-po)*5:y=7
860 IF  INKEY(0)>-1 AND po<20 THEN po=po+1 [2611]
:m=1
870 IF  INKEY(2)>-1 AND po>1 THEN po=po-1: [2462]
m=-1
880 IF  INKEY(18)>-1 THEN 960                [1082]
890 IF dm(po)=0 THEN po=po+m:IF po>20 THEN [6287]
 po=1:GOTO 890 ELSE IF po<1 THEN po=20:GOT
O 890 ELSE 890
900 LOCATE 67,4:GOSUB 1230                   [1144]
910 FOR a=1 TO 60:NEXT                       [1229]
920 a$=INKEY$:CLEAR INPUT:IF a$="" THEN 85 [1735]
0
930 IF a$="1" THEN dm(po)=2:GOTO 950         [2107]
940 IF a$="0" THEN dm(po)=1 ELSE GOTO 850    [2438]
950 LOCATE x,y:PRINT a$:GOTO 850             [1448]
960 a=128:FOR po=3 TO 10:IF dm(po)=2 THEN  [3595]
pb1=pb1+a
970 a=a/2:NEXT                               [1267]
980 a=128:FOR po=18 TO 11 STEP -1:IF dm(po [3139]
)=2 THEN pa1=pa1+a
990 a=a/2:NEXT                               [1267]
1000 IF dm(19)=2 THEN pa2=8                  [1409]
1010 IF dm(20)=2 THEN pa2=pa2+4              [1567]
1020 IF dm(2)=2 THEN pa2=pa2+2               [1465]
1030 IF dm(1)=2 THEN pa2=pa2+1               [905]
1040 OUT &F9F0,pa1+im1:OUT &F9F1,pb1+im2:O [1613]
UT &F9F4,pa2+im3+in
1050 POKE adr%,pa1+im1:POKE adr%+1,pb1+im2 [4260]
:POKE adr%+2,pa2+im3+in
1060 POKE adr%+3,INP(&F9F0):POKE adr%+4,IN [6504]
P(&F9F1):POKE adr%+5,INP(&F9F4):adr%=adr%+
6:byt=byt+6:NEXT
1070 CLG:LOCATE 15,7:PRINT SPACE$(50):LOCA [7842]
TE 15,8:PRINT SPACE$(50):LOCATE 15,18:PRIN
T SPACE$(50):LOCATE 15,19:PRINT SPACE$(50)
1080 LOCATE 20,24:PRINT SPACE$(40):NEXT      [1883]
1090 LOCATE 20,17:PRINT"Einprogrammierter  [4169]
IC absichern (J/N)"
1100 a$=INKEY$:IF a$="" THEN 1100            [1460]
1110 IF a$="n" OR a$="N" THEN END            [1541]
1120 POKE lgadr%,byt:POKE an%,PEEK(an%)+1: [5267]
SAVE"testicda.bin",b,an%,ende%+4+byt-an%
1130 END                                      [110]
1140 LOCATE x,y:PRINT CHR$(32):LOCATE x,y: [2858]
PRINT a$:RETURN
1150 IF INKEY(1)>-1 AND x<60 THEN x=x+5:xx [1947]
=xx+1:GOSUB 1220
1160 IF INKEY(8)>-1 AND x>15 THEN x=x-5:xx [4288]
=xx-1:GOSUB 1220
1170 IF INKEY(2)>-1 AND y=yi1 THEN y=y+yi3 [2505]
:yy=yy+1:GOSUB 1220
1180 IF INKEY(0)>-1 AND y=yi2 THEN y=y-yi3 [4108]
:yy=yy-1:GOSUB 1220
1190 IF INKEY(18)>-1 THEN e=1:RETURN         [2275]
1200 FOR p=1 TO 60:NEXT                      [1776]
1210 a$=UPPER$(INKEY$):IF a$="" THEN 1150  [2188]
ELSE RETURN
1220 LOCATE 67,4:IF yy=2 THEN po=xx ELSE p [2673]
o=21-xx
1230 PRINT"Position:";po:RETURN              [1884]
1240 'IC-Daten                               [491]
1250 DATA 20,0,0,-20,20,0,0,20,20,0,0,-20, [2360]
20,0,0,20,20,0,0
1260 DATA -20,20,0,0,20,20,0,0,-20,20,0,0, [2485]
20,20,0,0,-20,20
1270 DATA 0,0,20,20,0,0,-20,20,0,0,20,20,0 [2326]
,0,-20,20,0,0,20
1280 DATA 20,0,0,-20,20,0,0,20,20,0,0,-20, [2360]
20,0,0,20,20,0,0
1290 DATA -20,20,0,0,20,20,0,0,88,-20,0,0, [2462]
20,-20,0,0,-20
1300 DATA -20,0,0,20,-20,0,0,-20,-20,0,0,2 [1911]
0,-20,0,0,-20
1310 DATA -20,0,0,20,-20,0,0,-20,-20,0,0,2 [1911]
0,-20,0,0,-20
1320 DATA -20,0,0,20,-20,0,0,-20,-20,0,0,2 [1911]
0,-20,0,0,-20
1330 DATA -20,0,0,20,-20,0,0,-20,-20,0,0,2 [1911]
0,-20,0,0,-20
1340 DATA -20,0,0,20,-20,0,0,-20             [1033]
1350 DATA -20,0,0,-32,8,0,0,-24,-8,0,0,-32 [993]
```

```
100 'TESTS.LDR                               [520]
110 'erzeugt TESTS.BIN                       [1258]
120 '(c)1992 Markus Mueller                  [806]
130 '   & CPC International                  [2118]
140 '                                         [117]
150 DATA 00,00,ED,53,00,A0,C9,01             [612]
160 DATA F2,F9,3E,03,ED,79,01,F3             [1112]
170 DATA F9,ED,79,01,F6,F9,ED,79             [1696]
180 DATA 01,F7,F9,ED,79,3E,CF,ED             [1263]
190 DATA 79,3E,58,ED,79,01,F5,F9             [662]
200 DATA 3E,85,ED,79,C9,00,00,00             [1660]
210 FOR adr=&A000 TO &A02D:READ d$           [1476]
220 POKE adr,VAL("&"+d$):NEXT                [1294]
230 SAVE"tests.bin",b,&A000,&2D              [1095]
```

Spaß mit BASIC für Profis

Autor: Peter Krizian
Verlag: IDEA
ISBN: 3-88793-070-3

In Peter Krizians "Spaß mit BASIC für Profis" geht es um die Sprache, die unser CPC oder PCW von Haus aus spricht. BASIC, eine der am weitesten verbreiteten Programmiersprachen, ist zwar auf beiden Rechnern in verschiedenen Dialekten umgesetzt, beherrscht jedoch alle Standardbefehle. So findet man bei genauer Betrachtung von Mallard- und Locomotive-BASIC immer mehr Kongruenzen.
Anhand anschaulicher Beispiellistings versteht es Peter Krizian, den Interessierten in die Programmierung in Standard-BASIC einzuführen.
Dies beschränkt sich jedoch nicht nur auf Ergebnisse, die auf dem Monitor erzielt werden, sondern macht auch vor der Druckersteuerung nicht halt. Witzige Grafiken von Computern lockern das Bild dieses Buches enorm auf und nehmen so die Angst vor dem reinen Lehrbuch, als das sich Spaß mit BASIC versteht.

rs



Programmieren mit C

Autor: Matthias Clauß, Günther Fischer
Verlag: VEB Verlag Technik Berlin

Die Zusammenführung beider deutscher Länder bringt auch dem Heimanwender so manchen Vorteil. Know-how in Sachen Programmierung weiß das vorliegende Buch aus dem Ost-VEB-Verlag Technik Berlin zu vermitteln. Sauber gegliedert, vermag es den Leser zu fesseln und mit Hintergrundinformationen, die gerade für Einsteiger ungeheuer wichtig sind, zu versorgen. Im Selbststudium kann so jeder den Einstieg in die Sprache C finden.
Obwohl eigentlich für das Betriebssystem UNIX und dessen C-Compiler entworfen, kann auch jeder CPC- und PCW-Benutzer anhand der abgedruckten Beispieldateien die Programmierung in C erlernen. Gibt es doch auch für diese Systeme sehr umfangreiche Implementationen von C.
Doch vor dem ersten Listing steht noch eine Menge Arbeit.
So werden zuvor noch die grundlegenden Sprachelemente behandelt. Anhand anschaulicher Beispiele wird nun jedoch Zug um Zug die Verknüpfung mehrerer Befehle erörtert, um später ganze Listings anzubieten.

rs



BasiCode

Autor: Prof. Dr. rer. nat. Horst Völz
Verlag: Verlag Technik, Berlin
ISBN: 3-341-00895-0

Daß sich logische Zusammenhänge wesentlich besser begreifen lassen, wenn man selber in die Arbeit einbezogen wird und eventuell sogar bestimmte Punkte ausprobieren kann, hat sich inzwischen schon fast als Binsenweisheit durchgesetzt.
So wird zum Beispiel in der Schule nicht nur stur EDV-Stoff gepaukt, sondern es stehen auch Computer bereit, an denen man zuvor Besprochenes ausprobieren kann. Was aber tun, wenn man einen Computer zu Hause hat und nicht über einen Lehrer verfügt? Ganz einfach, man kauft sich ein Buch und versucht alles, was darin steht, auch genau auszuprobieren. Erste Programme werden geschrieben, und irgendwann trifft man dann doch mit anderen Computerneulingen zusammen, und siehe da, sie haben ein anderes Computersystem, eine ganz andere Programmiersprache.
Hier griff in den neuen Bundesländern ein ganzes Kollektiv ein, welches von einem Projekt in den Niederlanden erfahren hatte. Dort wurde eine Programmiersprache entwickelt, die auf unterschiedlichen Computersystemen lief, jedoch auf jedem System gleich zu programmieren war.
BasiCode konnte sich zwar in den Niederlanden nie richtig durchsetzen, die neuen Bundesländer indes nahmen die Idee dankbar auf.
So tat sich schon erwähntes Autorenkollektiv zusammen und sammelte sämtliche BasiCode-Interpreter, ergänzte sie und konnte dann ein vollständiges Lehrbuch mit Datenplatte veröffentlichen. Auf der beiliegenden Schallplatte befinden sich die BasiCode-Versionen für mehrere Rechnersysteme. Unter anderem auch für die CPC-Computer. Zur Programmierung gibt das Handbuch wichtige Tips, und anhand von kleineren und später auch großen Beispiellistings lernt sich die BasiCode-Sprache wie von selbst.

rs



Wir sind Ihr starker
CPC, Joyce & PC Partner



CPC 6128, der

ideale Computer für den Einsteiger:

mit Grünmonitor GT65 798,--
mit Farbmonitor CTM 644 1098,--
CPC 6128 Konsole einzeln 698,--
Grünmonitor GT 65 198,--
Farbmonitor CTM 644 398,--

Extrem preiswert: Amstrad CPC 464 aus Konkursmasse

Orginalverpackt, mit voller Garantie !

CPC 464 mit Farbmonitor CTM 644 und 9 Kassettenspielen DM 478,--

Farbmonitor CTM 644 (einzeln) **DM 398,--**



Amstrad CPC 464/6128 Plus

Idealer Einstiegscomputer !
Z80 A Prozessor • 128 KB Rom Cartridge mit AMSDOS • Basic 1.1 und Action - Rennspiel Burmin Rubber • DMA - Soundchip •

CPC 464 Plus mit 12" Monochrom Monitor **DM 599,-**
CPC 464 Plus mit 14" Farbmonitor **DM 799,-**
CPC 6128 Plus mit 12" Monochrom Monitor **DM 799,-**
CPC 6128 Plus mit 14" Farbmonitor **DM 1098,-**



vom Profi für den Profi:

dBaseII CPC / PCW 148,–
Wordstar 3.0 CPC / PCW 99,–
Multiplan CPC / PCW .... 99,– / 148,–
Basic/ Assembler CPC / PCW .... 99,– / 148,–
(alle Programme inkl. deutschem Handbuch)
Super !!
dBaseII, Wordstar, Multiplan auch im Paket erhältlich .... **nur 298,–**
Handbuch auch einzeln erhältlich 49,–



Grafik auf CPC:

EASIART + Trackerball 198,--
Zeichnen und Konstruieren mit dem Marconi Trackerball und dem Grafikprogramm EASI-ART. Unschlagbar in Kombination mit StopPress II
EASIART+ Trackerball +StopPress .. 348,--
EASIART Softw. u. Handb. 78,--

CPC

Arnor:

Arnor C CPC 6128 225,–
Prowort 198,-
MAXAM II 239,–

STAR DIVISION:

Star-Writer I 98,–
Datei-Star 98,–
Statistic-Star 98,–
FibuStar Plus CPC .. 298,–

CPC-Knüller:

ROMBOX CPC 118,–
MICA CPC 98,–
Terminalstar 3" 29,80
Mini Office II 3" 98,–
Dart-Scanner (+) 249,–
Videodigitizer (+) 348,–
Turbo Pascal 3.0 225,–
(+)Adapter für CPC 6128 39,–
Supercalc nur 6128 .. 59,80
HiSoft Pascal
Kas. / 3" .... 39,80 / 59,80
Terramaster 3" 49,– / 59,–
Mastercopy, 3"-CPC...65,–
Supercopy 65,–
Neuheit: Disktool 79,–
BTX Modul 198,–
MP3 199,–

CT1 79,–
Multiface II 178,–

Kassettensoftware:

Easi-Topcalc Kas. 29,80
Basic Lehrbuch Kass . 29,80
Mini Office II Kas. 49,–

Adventures CPC

(t= Text, g= Grafik)
Diamant von Rabenfels (g)
Drachenland (t)
Reise durch die Zeit (t)
Sherlock Holmes (g, nur 3")
Auftrag in der Bronx (g)
Insel der Smaragde (t)
Pharaonengrab (t)

je 3" Diskette 38,–
je Kassette 28,–

CPC + Joyce

DATENREM 68,–
ADRESSCOMP 58,–
FAKTUREM 78,–
FIBUPLAN 148,–
ETATGRAF 58,–
LAGDAT 68,–
KALKUREM 78,–
COMFORM 48,–
FIBUCOMP 98,–
PROFIREM 138,–

Grafik Knüller:

AMX Maus (Joyce)
mit Grafiksoftware 248,–
StopPress, das
DTP Programm 178,–
StopPress +
AMX Maus (Joyce) ... 298,–
zu StopPress CPC:
Fonts + Clipart 98,–
Vokabel/Verbentrainer 59,–

Handbücher Deutsch

für CPC:
Mini Office II 29,80
AMX-Maus CPC 29,80
StopPress CPC 19,80
dk'tronics Erw. 19,80
MAXAM 19,80
Supercalc 19,80
6128 deutsch 68,–
6128 original engl. ... 10,–
464 deutsch 48,–
DMP 2/3/4xxx 38,–
für Joyce:
AMX-Maus Joyce 19,80
Mini Office Prof. 29,80
StopPress Joyce 19,80
Desktop Publisher 19,80
LocoScript2 dt. 59,–
LocoFile dt. 59,–
LocoSpell dt. 59,–

Wir haben die Lizenz für Kotulla PD-Programme (CPC u. Joyce)

Preiswerte Programme für **CPC** und **Joyce** - so macht Software Spaß !

1 **JRT-Pascal** - vollständiger Pascal-Compiler *
2 **Z80-Assembler**, Disassembler, Linker, Debugger
3 Künstliche Intelligenz - **XLISP** u. **E-PROLOG** *
4 C-Compiler **Small-C** - mit Fließkommazahlen *
5 **FORTH-83** - Komfortabler Forth-Interpreter
6 **Utilities**: Dateikompression, Diskmon., Dateiretter...
7 Programme aus dem **Großen CPC-Arbeitsbuch** **
8 Adventure **Colossal Cave** (Pg. engl.) *
9 **Disk Utilities** - kopiert geschützte Software **
10 **BizBasic** - Umfangreiche Basic-Erweiterung **
11 Basic-Compiler **E-BASIC** für CP/M
12 **Turbo Pascal-Programme** - Turbo-Inliner, Grafik
13 Programme aus **Den Joyce programmieren**
14 Programme aus **CPC-Dateiverwaltung** **
15 **WordStar-Tools** - Fußnoten, Index, Spaltendruck *
16 **dBASE-Literarurverwaltung** *
17 C-Interpreter **SCI** - Spielend C lernen *
18 **MacroPack/Z80** - Makroassembler, Debugger, Linker
19 **DFÜ-Programm** MEX - Datenübertragung *
20 **WS-Tuner**

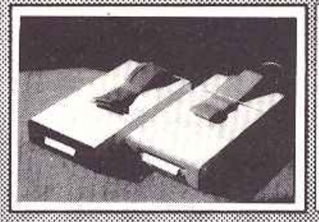
* Auf CPC-464/664 nur mit Speichererweiterung (min. 64 K) ** Nicht geeignet für Joyce
Mit deutschem Handbuch

nur DM 25,-- pro 3"-Diskette **DM 60,--** für drei beliebige 3"-Disketten



AMSTRAD 3"-Laufwerke:

DD1 (inkl. Contr.) 398,--
Ein Muß für jeden CPC 464 -Besitzer
Controller (auch einzeln erhältlich) **198,–**
FD1 (2.-Laufwerk CPC) 3" **198,--**
Achtung! Läuft am CPC 464 nur mit Contr.
Kabel für FD1 nötig für 6128 / 664 **48,–**
FD4 (2.-EinbauLaufw. Joyce) 3" **398,–**



Zweit-Laufwerke / Festplatten

3,5" LW (CPC / Joyce) .... **je 240,–**
Achtung! Läuft am CPC 464 nur mit Contr.
5,25" LW (CPC / Joyce) .... **je 320,–**
Achtung! Läuft am CPC 464 nur mit Contr.
Diskpara 3,5" / 5,25 " Formatsoftware 78,–

HD 20 (CPC 464/664/6128) **1.100,--**
Joyce Festplatte 30 MB 798,--



Neuheiten CPC - Joyce

Joystick Controller Joyce 69,–
- Spielen Sie Ihre Joyce Spiele jetzt auch mit dem Joystick.
- **Joyce Adapter erforderlich** 39,–

CPC Schnittstelle 199,50
Unbedingt erforderlich zur DFÜ
Musizier mit mir 99,80
Neues Musikprogramm für den CPC !!

Joyce Software und Hardware

Joyce PCW 8256 998,–
Joyce PCW 8512 1398,–
Joyce PCW 9512 1698,–

Locomotive:

LocoMail 1 128,–
LocoMail 2 168,–
LocoScript 2 148,–
Loco Spell2 168,–
Loco File 168,–
Loco Font
Set 1 und 2 ...... 79,80 / 68,40
Prospell dt. 79,–

STAR DIVISION

Statistik-Star 98,–
Starmail 99,–
Datei-Star 99,–
Mailing-System 189,–
Star-Base 198,–
Business-Star 298,–

Fibu-Star Plus 298,–
Kontenblätter Fibu 35,90

Joyce-Spiele:

Anals of Rome 89,–
Batman 59,–
Cyrus 3D Schach 49,80
Ski Climb 79,–
Tetris 79,–
Domino,Backgammon . 69,–
Matchday II 69,–
Tomahawk 79,–

Joyce-Knüller:

Supercopy (Kopierprg.) ... 85,–
Desktop Publisher 98,–
... + AMX-Maus (+) 298,–
MasterScan (+) 298,–
VIDI PCW (+) 348,–
Echtzeituhr (+) 129,–
(+) PCW-Adapter 39,–

Mini Office Profess. .. 138,–
...Dtsch. Zeichensatz 29,80
FISKUS 91 139,–
WS-Tuner 49,80
Turbo Pascal 3.0 225,-
Prompt (Datei) 69,–
Prompt Druck 39,–
MICA CAD 98,–
Comac Litbox 4.0 148,–
Comac Kasse Plus ... 168,–
Comac-Banktransfer . 59,50
Vereinsverwaltung.... 198,–
Schreiblehrgang 89,–
Mini DOS 50,–
Lerntrainer Joyce 79,–
Auswahltrainer 49,–
Arnor C 225,–
Prowort 198,–
MAXAM II 199,–
ProScan (Handscanner) 798,–

Joyce + CPC Zubehör

Abdeckhauben:

CPC Monitore .... je 39,80
DMP-Drucker .... je 29,80
DD1/FD1/Vortex .... je 19,80
CPC Konsolen ...... je 24,80
Joyce Monitor 44,80
Joyce Tastatur 24,80
Joyce Drucker 24,80

3"-Disketten (10 St.)

Maxell / AmsoftCF2 59,–
CF2 DD 98,–
Noname 49,–

Diskettenboxen:

3"-3,5" /50 19,80
3"-3,5"/100 29,80
5,25"/100 29,80
POSSO BOX 150 3" ... 59,–

Diverses:

Datenrekorder + Kab. 98,–
Monitorverlängerung:
464 / 6128 ..... 24,50 / 29,80
Druckerkabel CPC 38,–
Traktor NLQ 401 58,–
Bildschirmfilter 39,–
Druckerverlängerung:
für Joyce (2 Kabel) 58,–
Papierführung 19,80

Joysticks

Competition pro 39,80
Quickshot II 19,80
Joystickadapter 39,80

Schnittstellen:

CPC Schnittstelle 199,50
RS 232 Joyce........ .. 198,–

Etiketten 200 Stk
70 x 70 3" und 3,5" 16,–

Farbbänder

Joyce 24,80
DMP 2xxx 19,80
NLQ 401 19,80

Schaltpläne

CPC Konsole .... je 29,80
CTM 644/640 19,80
GT 64/65 19,80
DDI1/FD1 19,80
DMP xxxx 29,80
Joyce 8256/8512 29,80
Speichererweiterung
64KB 149,–
256KB 269,–
256KB Joyce 128,–

Karl-Heinz Weeske
Potsdamer Ring 10
D-7150 Backnang
Kreissparkasse BK • BLZ (60250020)
74397 • Postgiro Stgt. 83326-707



COMPUTER-ELEKTRONIK

Zahlung per Nachnahme oder Vorauskasse. Versandkostenpauschale: Inland 9,80 DM (Ausland 19,80 DM)

Fax: 07191-60077
Tel.: 07191-1528(29), 60076

zurück an Absender 10/11 '92

Interessiert an weiterem Infomaterial ? Kostenlose Gesamtangebotslisten anfordern !

○ für CPC 6128, 664, 464
○ für PC ○ für Joyce
○ Spiele CPC ○ Spiele Joyce
○ Lernprogramm-Liste

Vorname, Name:
Straße, Hausnummer:
PLZ, Ort:
Telefon-Nr, Datum:
Mein Computersystem:



Besitzerfreuden

Serie: Der Weg zum eigenen Rollenspiel

Ein Abenteuerspiel, das wirklich Stimmung bringen soll, muß auch Gelegenheit bieten, die niederen Triebe zu befriedigen – zum Beispiel die Habgier. Das neueste Programmodul unserer Rollenspielserie erlaubt daher das Sammeln und Weitergeben von Objekten. Und da Gemeinheiten nur in Gesellschaft wirklich Spaß machen, sind jetzt auch erstmals mehrere Spieler mit dabei.

Willkommen zur vorletzten Folge unserer Serie! Mit dem "Creator 4" bekommen Sie diesmal ein erweitertes Rollenspiel-Rahmenprogramm an die Hand, das eine Besitztümer-Verwaltung und Eigenschaftskonten mitbringt. Außerdem können nun vier Spieler im Team agieren; erst dadurch kommt das klassische Rollenspiel-Feeling zustande.

Das neue Programm enthält bereits alle Funktionen der "Creator"-Module 2 und 3 und macht diese Programme somit überflüssig. Den Ur-"Creator" aus Heft 6/7'92 brauchen Sie allerdings wieder, um die Grafikpuzzle-Dateien für Sprites, Landkarte, Stadt- und Häuserpläne zu erzeugen.

Nach dem Start des Programms sieht man unterhalb des Schauplatzfensters vier Spielerfiguren. Der grün dargestellte Kopf steht für den gerade aktiven Spieler. Im Schauplatzfenster erkennt man die aktive Spielerfigur daran, daß sie transparent dargestellt ist. Durch Druck auf die Leertaste wechselt die aktive Spielerfigur. Die Taste <S> ermöglicht eine Identifikation von Gegenständen, auf denen die aktive Spielerfigur gerade steht. Das ist nötig, denn immerhin kommen jetzt ja Objekte aller Art ins Spiel.

Heldenquartett

Mit <Return> gelangt man in das persönliche Statusmenü der aktiven Figur. Hier werden die aktuellen Werte der Persönlichkeitskonten angezeigt. Außerdem findet hier das Aufnehmen, Ablegen und Weitergeben von Objekten statt.

Der Menüpunkt "Benutzen" ist bereits vorgesehen, ohne daß hierfür bislang irgendwelche Reaktionen programmiert wären. Es hängt von der konkreten Spielstory ab, welche Gegenstände bei welchen Figurenpositionen "benutzt" werden müssen, um den Wert eines bestimmten Kontos zu verändern oder andere Reaktionen auszulösen. Eine entsprechende Abfrage muß in der Routine ab Zeile 2030 verankert werden (IF A$="B" THEN...).

Das Umschalten zwischen den Spielerfiguren wird von Zeile 280 bis 480 bewerkstelligt. Dabei wird jedesmal kontrolliert, ob die nächste Figur nicht bereits tot ist. Die Zeilen 390 bis 440 dienen dazu, den neuen Schauplatzausschnitt so zu wählen, daß sich die aktive Spielerfigur immer in der Mitte befindet.

In den Zeilen 1860 bis 1890 wird die Anzeige der Spielerfigur aufgebaut. Das Ende des Spiels ist erreicht, wenn alle vier Spielerfiguren tot sind. Die einzige Todesgefahr, die das Rahmenprogramm bisher bietet, stellt eine Kollision mit dem sich ausbreitenden Feuer im linken oberen Bereich der Landkarte dar.

Durch die Kollisionsabfrage und die Lebenskraft-Konten ist das Programm jedoch auch für alle anderen denkbaren "Todesarten" offen. Die Liste der Ge-

Maschinenprogrammaufrufe

Das Maschinenprogramm zur neuen "Creator"-Version wurde gegenüber der Version aus dem letzten Heft nur geringfügig verändert. Assemblerfreunde finden den kompletten Quelltext auf der DATABOX-Diskette zu diesem Heft.

Das Maschinenprogramm wird von Basic aus mit folgenden CALLs angesprochen:

A000,adr:	initialisiert die Anfangsadresse, an welche die Daten der Grafikpuzzles geladen worden sind.
A1fe,adr:	invertiert die Spielerfiguranzeige an Bildschirmadresse "adr"
A17c,feld,aus	Darstellung des Landschaftsausschnitts mit der Anfangsadresse "feld" und der Ausdehnung "aus"
A1ef,adr,adr2	transparente Darstellung der aktiven Spielerfigur mit der Grafikadresse "adr2" an die Bildschirmadresse "adr"
A156,adr,adr2	Überschreiben durch Puzzle mit der Grafikadresse "adr2" an die Bildschirmadresse "adr"
A00a,gra,h ,feld,aus,cur	scrollt den Landschaftsausschnitt nach rechts, dabei sind: gra = Adresse des Grafikpuzzles der Spielerfigur h = Nummer des Hintergrundpuzzles der alten Position feld = Adresse der Figur in der Ebene aus = Ausdehnung der aktuellen Ebene cur = Bildschirmadresse der aktuellen Figur
A048,gra,h, feld,aus,cur	scrollt den Landschaftsausschnitt entsprechend nach links
A086,gra,h, feld,0,cur	scrollt den Landschaftsausschnitt nach unten. Parameter wie beim Scrollen nach rechts, jedoch ohne Angabe der Ausdehnung (dafür 0)
A0c4,gra,h, feld,0,cur	scrollt den Landschaftsausschnitt entsprechend nach oben

Neue Variablen

figadr(1) bis figadr(4):	Bildschirmadressen der vier Köpfe (Spielerfigurenanzeige)
aktfig	Nummer der aktiven Spielerfigur
pirat(n,6):	Zur Kontrolle jeder Spielerfigur sind 6 Parameter nötig, die zu Anfang aus den DATA-Zeilen ab 1060 gelesen werden. Sie stehen dort in der Reihenfolge "DATA s,e,g,x,y,h" und haben im einzelnen folgende Bedeutung: s = Status (1: lebend, 0: tot) e = Ebene, in der sich die Figur momentan befindet g = Nummer des Grafikpuzzles, das die Figur darstellen soll x = X-Koordinate auf der Landkarte, "1,1" = linke obere Ecke) y = Y-Koordinate h = Nummer des Hintergrundgrafikpuzzles
name2$(n,2):	In dieser Variablen stehen Name und Geschlecht der jeweiligen Spielerfigur, die dazugehörigen DATAs sind in Zeile 1930.
werte(n,12):	Die 12 Eigenschaftswerte einer jeden Spielerfigur werden aus den DATA-Zeilen ab 1940 in diese Variable gelesen.
gegenst$(14):	In den Zeilen 1100 und 1110 stehen die Namen der Gegenstände, die in diese Variable eingelesen werden.
besitz(n,6):	beschreibt den Inhalt von einer der 6 Taschen einer Figur in Form einer Nummer, die sich auf den Index im Feld "gegenst$" bezieht.
spieler:	Anzahl der noch lebenden Spielerfiguren
tabelle:	Beginn der Tabelle, in der die Parameter der abgelegten Gegenstände stehen. Es werden folgende Parameter gesichert: 1. die jeweilige Ebene 2. X-Koordinate 3. Y-Koordinate 4. Nummer des Gegenstands (wie Index "gegenst$") 5. Nummer des Hintergrundgrafikpuzzles
last(1) bis last(4):	Tabelle der Ebenen-Rücksprungdaten der jeweiligen Spielerfigur
invert:	Adresse der Invertierungsroutine
Temporäre Variablen:	
tnr:	Tasche, aus der man etwas ablegen will
adr:	Speicheradresse der aktuellen Spielerfigur auf der Landkarte
tnr2:	Tasche, in der man etwas hineinlegen will
spnr:	Nummer der Figur, der man etwas übergeben will
status(1) bis status(4):	zur Ermittlung, ob und welche Spielerfiguren die aktive berühren

genstände kann ebenso wie das vorhandene Sortiment an Feind-Charakteren nur ein Anfang sein. Wenn Sie eine der Listen erweitern, achten Sie bitte auf die richtige Anfangszeile nach dem jeweiligen RESTORE-Befehl. Werden Ebenen beim Betreten nachgeladen, sind sie auch beim Verlassen wieder abzuspeichern – damit die jeweils erfolgten Änderungen während des Spielens in der betreffenden Ebene auch erhalten bleiben. Im abgedruckten Beispiel haben wir das Abspeichern noch nicht realisiert. Hier werden die Beispielebenen gleich zu Beginn des Spiels geladen und bleiben fortan resident im Speicher.

Matthias Fink/Ulrich Schmitz/sz

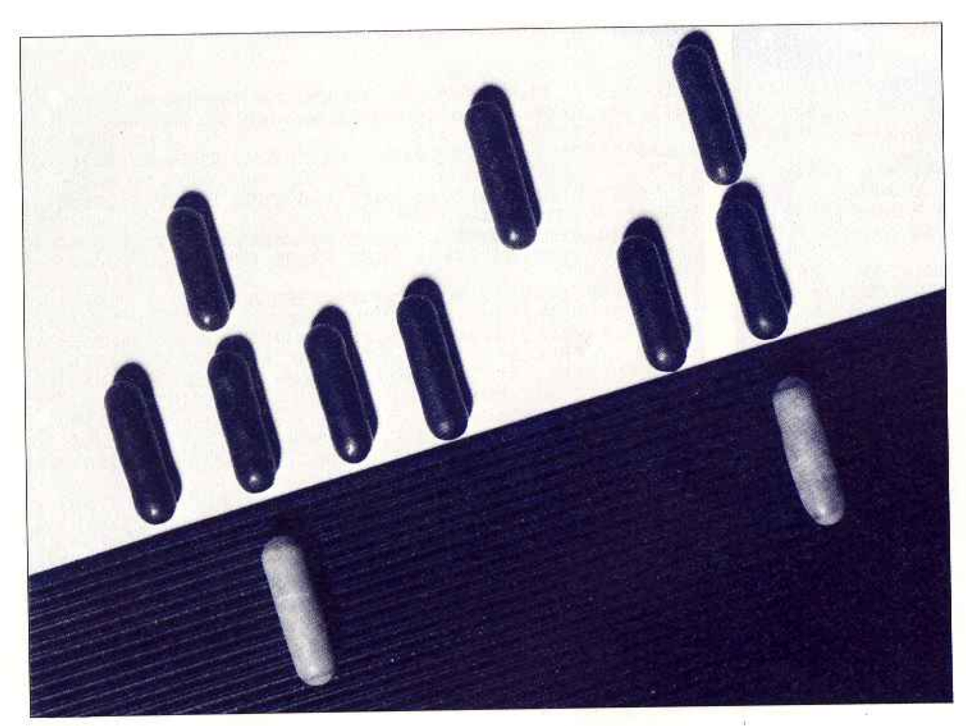
```
10 'CREATOR4.BAS (c) 1992 M. Fink & CPC In [2181]
t.
20 DATA 18,2,0,7,23,12,18,9,24,15,6,3,8,13 [1201]
,10,26
30 FOR a=0 TO 15:READ n:INK a,n:NEXT:BORDE [2496]
R 0:MODE 0:PAPER 2:CLS
40 RESTORE 950:pmax=5:ind=6:DIM m(pmax),xa [9357]
lt(pmax),yalt(pmax),name$(pmax),xaus(pmax)
,yaus(pmax),spec(pmax):DIM s(ind),e(ind),x
k(ind),yk(ind),t(ind),a(ind),h(ind),z(ind)
50 FOR n=1 TO pmax:READ m(n),xalt(n),yalt( [7564]
n),name$(n),xaus(n),yaus(n),spec(n):NEXT
60 RESTORE 1000:FOR n=1 TO ind:READ s(n),e [4506]
(n),xk(n),yk(n),t(n),a(n),h(n),z(n):NEXT
70 ebene=1:aktfig=1:figadr(1)=&C69B:figadr [6054]
(2)=&C6AB:figadr(3)=&C6BB:figadr(4)=&C6CB
80 DIM pirat(4,6):RESTORE 1060:FOR m=1 TO  [3106]
4:FOR n=1 TO 6:READ pirat(m,n):NEXT:NEXT
90 DIM werte(4,12),name2$(4,2),gegenst$(14 [3913]
),besitz(4,6):RESTORE 1930
100 FOR n=1 TO 4:READ name2$(n,1):READ nam [2647]
e2$(n,2):NEXT
110 FOR n=1 TO 4:FOR m=1 TO 12:READ werte( [4083]
n,m):NEXT:NEXT
120 RESTORE 1100:FOR n=0 TO 14:READ gegens [2895]
t$(n):NEXT
130 FOR n=1 TO 4:FOR m=1 TO 6:READ besitz( [3633]
n,m):NEXT:NEXT
140 MEMORY 21999:LOAD"creator4.bin",&A000: [3115]
CALL &A000,23470
150 LOAD"demo4.spr",23469:LOAD"demo4.fld", [4493]
29300
160 LOAD"stadt1.fld",31800:LOAD"stadt1.fld [7940]
",32425:LOAD"stadt1.fld",33050:LOAD"haus1.
fld",33675
170 spieler=4:feld=29300:tabelle=22000:las [8410]
t(1)=23000:last(2)=23100:last(3)=23200:las
t(4)=23300:frite1=&A156:frite=&A1EF:indnr=
1:invert=&A1FE
180 GOSUB 390:GOSUB 1860:GOSUB 1830        [2611]
190 a$=INKEY$:IF a$="" THEN a$="@"         [1910]
200 b=ASC(a$):IF b=11 THEN GOSUB 610       [1405]
210 IF b=10 THEN GOSUB 670                 [1743]
220 IF b=8 THEN GOSUB 490                  [579]
230 IF b=9 THEN GOSUB 550                  [1317]
240 IF b=32 THEN GOSUB 280                 [979]
250 IF b=13 THEN GOSUB 1980                [2083]
260 IF b=115 THEN GOSUB 2640               [1497]
270 GOSUB 1160:GOTO 190                    [1211]
280 CALL invert,figadr(aktfig)             [2621]
290 pirat(aktfig,2)=ebene:pirat(aktfig,4)= [4181]
x+xscr:pirat(aktfig,5)=y+yscr
300 pirat(aktfig,6)=PEEK(feld+xscr+x+(yscr [3650]
+y)*xaus(ebene))
310 POKE (feld+xscr+x+(yscr+y)*xaus(ebene) [2469]
),pirat(aktfig,3)
320 'hintergrund                           [339]
330 aktfig=aktfig+1:IF aktfig=5 THEN aktfi [2762]
g=1
340 IF pirat(aktfig,1)=0 THEN 330          [1920]
350 IF pirat(aktfig,2)=ebene THEN 390      [3243]
360 ebene=pirat(aktfig,2):n=VAL(name$(eben [2358]
e))
370 IF n<>0 THEN feld=n:GOTO 390           [1321]
380 LOAD name$(ebene)+".fld",29300:feld=29 [2641]
300
390 xscr=pirat(aktfig,4)-4:yscr=pirat(aktf [3997]
ig,5)-4:x=4:y=4
400 IF xscr<0 THEN x=x+xscr:xscr=0         [2927]
410 IF yscr<0 THEN y=y+yscr:yscr=0         [1971]
420 IF xscr>xaus(ebene)-10 THEN x=x+xscr-( [5163]
xaus(ebene)-10):xscr=xaus(ebene)-10
430 IF yscr>yaus(ebene)-10 THEN y=y+yscr-( [5347]
yaus(ebene)-10):yscr=yaus(ebene)-10
440 POKE (feld+x+xscr+(y+yscr)*xaus(ebene) [4951]
),pirat(aktfig,6)
450 CALL invert,figadr(aktfig)             [2621]
```

```
460 curgra=23470+pirat(aktfig,3)*80:cur=49 [2686]
167+x*5+y*160
470 CALL &A17C,feld+xscr+yscr*xaus(ebene), [6407]
xaus(ebene):CALL frite,cur,curgra
480 LOCATE 5,21:PRINT SPACE$(13):LOCATE 5, [4474]
21:PRINT name2$(aktfig,1):RETURN
490 IF x=1 AND xscr=0 THEN RETURN:'links   [1836]
500 a=PEEK(feld+xscr+yscr*xaus(ebene)+x+y* [4294]
xaus(ebene)-1):GOSUB 730
510 IF a=256 THEN RETURN                   [1275]
520 h=PEEK(feld+xscr+yscr*xaus(ebene)+x+y* [3634]
xaus(ebene))
530 IF x=1 THEN xscr=xscr-1:CALL &A00A,pir [6537]
at(aktfig,3)*80+23470,h,feld+xscr+yscr*xau
s(ebene),xaus(ebene),cur:RETURN
540 x=x-1:CALL frite1,cur,h*80+23470:cur=c [6882]
ur-5:CALL frite,cur,curgra:RETURN
550 IF x=8 AND xscr=xaus(ebene)-10 THEN RE [1391]
TURN:'rechts
560 a=PEEK(feld+xscr+yscr*xaus(ebene)+x+y* [4172]
xaus(ebene)+1):GOSUB 730
570 IF a=256 THEN RETURN                   [1275]
580 h=PEEK(feld+xscr+yscr*xaus(ebene)+x+y* [3634]
xaus(ebene))
590 IF x=8 THEN xscr=xscr+1:CALL &A048,pir [9364]
at(aktfig,3)*80+23470,h,feld+xscr+yscr*xau
s(ebene)+9,xaus(ebene),cur:RETURN
600 x=x+1:CALL frite1,cur,h*80+23470:cur=c [5936]
ur+5:CALL frite,cur,curgra:RETURN
610 IF y=1 AND yscr=0 THEN RETURN:'oben    [850]
620 a=PEEK(feld+xscr+yscr*xaus(ebene)+x+(y [5386]
-1)*xaus(ebene)):GOSUB 730
630 IF a=256 THEN RETURN                   [1275]
640 h=PEEK(feld+xscr+yscr*xaus(ebene)+x+y* [3634]
xaus(ebene))
650 IF y=1 THEN yscr=yscr-1:CALL &A086,pir [8938]
at(aktfig,3)*80+23470,h,feld+xscr+yscr*xau
s(ebene),0,cur:RETURN
660 y=y-1:CALL frite1,cur,h*80+23470:cur=c [6662]
ur-160:CALL frite,cur,curgra:RETURN
670 IF y=8 AND yscr=yaus(ebene)-10 THEN RE [3905]
TURN:'unten
680 a=PEEK(feld+xscr+yscr*xaus(ebene)+x+(y [4461]
+1)*xaus(ebene)):GOSUB 730
690 IF a=256 THEN RETURN                   [1275]
700 h=PEEK(feld+xscr+yscr*xaus(ebene)+x+y* [3634]
xaus(ebene))
710 IF y=8 THEN yscr=yscr+1:CALL &A0C4,pir [6799]
at(aktfig,3)*80+23470,h,feld+xscr+yscr*xau
s(ebene)+9*xaus(ebene),0,cur:RETURN
720 y=y+1:CALL frite1,cur,h*80+23470:cur=c [4852]
ur+160:CALL frite,cur,curgra:RETURN
730 'ereignisabfrage                       [562]
740 IF a=42 THEN RETURN                    [692]
750 IF a=4 OR a=5 OR a>37 THEN a=256:RETUR [1838]
N
760 IF a>30 THEN lx=x+xscr:ly=y+yscr:GOTO  [2263]
1690:'Kollision
770 IF a<>25 THEN 840                      [1296]
780 last(aktfig)=last(aktfig)-5            [1880]
790 n=PEEK(last(aktfig)):x=PEEK(last(aktfi [5889]
g)+1):y=PEEK(last(aktfig)+2):xscr=PEEK(las
t(aktfig)+3):yscr=PEEK(last(aktfig)+4)
800 name=VAL(name$(n)):IF name<>0 THEN fel [4115]
d=name:GOTO 820
810 LOAD name$(n)+".fld",29300:feld=29300  [2499]
820 cur=49167+5*x+160*y                    [1604]
830 ebene=n:CALL &A17C,feld+xscr+yscr*xaus [5611]
(ebene),xaus(ebene):CALL frite,cur,curgra:
a=256:RETURN
840 FOR n=1 TO pmax                        [1092]
850 IF a<>m(n) THEN 940                    [562]
860 IF ABS(xalt(n)-(x+xscr))>1 THEN 940    [1325]
870 IF ABS(yalt(n)-(y+yscr))>1 THEN 940    [1931]
880 POKE last(aktfig),ebene:POKE last(aktf [7058]
ig)+1,x:POKE last(aktfig)+2,y:POKE last(ak
tfig)+3,xscr:POKE last(aktfig)+4,yscr
```

```
890 ebene=n:last(aktfig)=last(aktfig)+5   [1772]
900 name=VAL(name$(n)):IF name<>0 THEN fel [4127]
d=name:GOTO 920
910 LOAD name$(n)+".fld",29300:feld=29300  [2499]
920 x=4:y=4:xscr=0:yscr=0:cur=49167+5*x+16 [1127]
0*y
930 n=pmax:a=256:CALL &A17C,feld+xscr+yscr [6145]
*xaus(ebene),xaus(ebene):CALL frite,cur,cu
rgra:GOSUB 1830
940 NEXT:RETURN                            [940]
950 DATA 255,0,0,29300,50,50,0             [572]
960 DATA 20,33,21,31800,25,25,0            [1466]
970 DATA 20,21,28,32425,25,25,0            [876]
980 DATA 20,44,27,33050,25,25,0            [1324]
990 DATA 22,2,1,33675,15,15,0              [1015]
1000 DATA 1,1,27,24,1,31,0,0               [1169]
1010 DATA 1,1,49,49,2,32,0,0               [868]
1020 DATA 1,1,38,4,3,32,1,0                [1095]
1030 DATA 1,2,9,9,3,31,28,0                [630]
1040 DATA 1,3,22,22,2,32,28,0              [862]
1050 DATA 1,1,9,4,4,33,33,5                [1379]
1060 DATA 1,1,38,4,4,1                     [917]
1070 DATA 1,1,39,21,30,1                   [975]
1080 DATA 1,1,40,44,25,1                   [926]
1090 DATA 1,1,41,31,22,0                   [890]
1100 DATA ,Lampe,Messer,Kerze,Oel,Dolch,Sa [5449]
ebel,Seil,Hammer,Meissel,Naegel
1110 DATA Zunder,Dietrich,Schluessel,Rucks [2899]
ack
1120 DATA 1,2,0,0,0,0                      [564]
1130 DATA 0,0,3,6,9,0                      [896]
1140 DATA 10,11,0,0,0,0                    [578]
1150 DATA 0,4,0,13,2,0                     [846]
1160 n=indnr                               [509]
1170 IF s(n)=0 THEN 1500                   [1462]
1180 IF ebene<>e(n) THEN 1500:'Wegfall, we [4605]
nn "Hintergrundbewegung" erwuenscht
1190 lx=xk(n):ly=yk(n)                     [545]
1200 ON t(n) GOTO 1210,1280,1350,1400      [2120]
1210 IF z(n)>39 THEN z(n)=0:'Taktik1, Rech [2463]
teckabschritt
1220 lnr=INT(z(n)/10)+1                    [1441]
1230 ON lnr GOTO 1240,1250,1260,1270       [1884]
1240 lx=lx-1:GOTO 1490                     [1598]
1250 ly=ly+1:GOTO 1490                     [1619]
1260 lx=lx+1:GOTO 1490                     [1158]
1270 ly=ly-1:GOTO 1490                     [1571]
1280 l1=x+xscr-xk(n):l2=y+yscr-yk(n):'Takt [5030]
ik2, Spielerverfolgung
1290 IF l1=0 THEN 1310                     [613]
1300 l1=l1/ABS(l1)                         [594]
1310 IF l2=0 THEN 1330                     [364]
1320 l2=l2/ABS(l2)                         [573]
1330 lx=lx+l1:ly=ly+l2                     [1771]
1340 GOTO 1490                             [313]
1350 ON INT(RND*4+1) GOTO 1360,1370,1380,1 [3059]
390:'Taktik3, Zufallsbewegung
1360 lx=lx-1:IF lx<0 THEN 1500:ELSE GOTO 1 [1837]
490
1370 lx=lx+1:IF lx=xaus(ebene) THEN 1500:E [2061]
LSE GOTO 1490
1380 ly=ly-1:IF ly<0 THEN 1500:ELSE GOTO 1 [2259]
490
1390 ly=ly+1:IF ly=yaus(ebene) THEN 1500:E [2330]
LSE GOTO 1490
1400 z(n)=z(n)-1:IF z(n)>0 GOTO 1500:'Takt [2705]
ik4, Ausbreitung
1410 lnr=INT(RND*4+1)                      [960]
1420 ON lnr GOTO 1430,1440,1450,1460       [2462]
1430 lx=lx-1:IF lx<0 THEN 1500:ELSE GOTO 1 [1809]
470
1440 lx=lx+1:IF lx=xaus(ebene) THEN 1500:E [1969]
LSE GOTO 1470
1450 ly=ly-1:IF ly<0 THEN 1500:ELSE GOTO 1 [2279]
470
1460 ly=ly+1:IF ly=yaus(ebene) THEN 1500   [2291]
1470 lnr=PEEK(feld+lx+ly*xaus(ebene)):IF l [3718]
nr=a(n) THEN 1500
1480 h(n)=a(n):z(n)=5:GOTO 1490            [1770]
1490 GOSUB 1520                            [907]
1500 indnr=indnr+1:IF indnr>ind THEN indnr [2116]
=1
1510 RETURN                                [555]
1520 l1=PEEK(feld+lx+ly*xaus(ebene))       [2511]
1530 IF l1>37 THEN 1730:'Kollision mit ein [3968]
er Spielerfigur
1540 IF lx=x+xscr AND ly=y+yscr THEN l1=ak [5333]
tfig+37:GOTO 1730:'Kollision mit akt. Spie
lerfigur
1550 IF l1>29 THEN RETURN                  [1256]
1560 IF l1<28 AND l1>21 THEN RETURN        [1241]
1570 IF l1=20  OR l1=17 OR l1=16 THEN RETU [2854]
RN
1580 IF l1<7 AND l1>3 THEN RETURN          [968]
1590 l1=xk(n)-xscr:l2=yk(n)-yscr           [1818]
1600 IF l1>9 OR l2>9 OR l1<0 OR l2<0 THEN  [2365]
1620
1610 CALL fritel,49167+l1*5+l2*160,23470+h [2420]
(n)*80
1620 l1=lx-xscr:l2=ly-yscr                 [1428]
1630 IF l1>9 OR l2>9 OR l1<0 OR l2<0 THEN  [2383]
1650
1640 CALL frite,49167+l1*5+l2*160,23470+a( [1287]
n)*80
1650 z(n)=z(n)+1:POKE feld+xk(n)+yk(n)*xau [3242]
s(ebene),h(n)
1660 h(n)=PEEK(feld+lx+ly*xaus(ebene))     [1300]
1670 POKE feld+lx+ly*xaus(ebene),a(n)      [2669]
1680 xk(n)=lx:yk(n)=ly:RETURN              [3162]
1690 IF a=33 THEN l1=aktfig+37:GOTO 1770   [1801]
1700 FOR n=1 TO ind                        [797]
1710 IF ABS(lx-xk(n))<2 AND ABS(ly-yk(n))< [3080]
2 AND e(n)=ebene THEN 1730
1720 NEXT n                                [366]
1730 IF a(n)=33 THEN 1770                  [806]
1740 l1=xk(n)-xscr:l2=yk(n)-yscr:s(n)=0:a= [2218]
256
1750 CALL fritel,49167+l1*5+l2*160,23470+h [2420]
(n)*80
1760 POKE feld+xk(n)+yk(n)*xaus(ebene),h(n [4056]
):PRINT CHR$(7):BORDER 0:RETURN
1770 CLS:pirat(l1-37,1)=0:POKE feld+pirat( [4038]
l1-37,4)+pirat(l1-37,5)*xaus(ebene),pirat(
l1-37,6)
1780 LOCATE 1,10:PRINT"Die Spielerfigur":P [7787]
RINT:PRINT"mit der Nummer ";l1-37:PRINT:PR
INT"ist tot. (Taste)":CALL &BB18
1790 spieler=spieler-1:IF spieler>0 THEN 1 [2524]
810
1800 CLS:LOCATE 1,10:PRINT"Die ganze Grupp [5857]
e":PRINT:PRINT"ist tot. Spielende!":PRINT:
END
1810 CLS:IF (l1-37)=aktfig THEN GOSUB 330: [3324]
ELSE GOSUB 450
1820 GOSUB 1860:a=256:RETURN               [1927]
1830 FOR n=1 TO ind:IF e(n)<>ebene OR s(n) [2242]
=0 THEN 1850
1840 lx=xk(n):ly=yk(n):z(n)=z(n)-1:GOSUB 1 [2388]
620
1850 NEXT:RETURN                           [940]
1860 a=&C69B+2^16:FOR n=a TO a+48 STEP 16  [1660]
1870 IF pirat(((n-50843)/16)+1,1)=0 THEN 1 [1691]
890
1880 CALL fritel,n,23470+37*80:CALL fritel [6738]
,n+5,23470+36*80:CALL fritel,n+160,23470+3
4*80:CALL fritel,n+165,23470+35*80
1890 NEXT:CALL invert,figadr(aktfig):RETUR [1930]
N
1900 DATA Grad,Alter,Groesse (cm),Gewicht  [2792]
(kg)
1910 DATA Staerke,Geschicklichkeit,Konstit [4023]
ution,Intelligenz
1920 DATA Auftreten,Reaktionswert,Bewegung [3551]
,Lebenspunkte
1930 DATA Klaus,m,Sabine,w,Peter,m,Sybille [1738]
,w
1940 DATA 1,19,176,78,90,92,98,80,80,89,29 [2176]
,40
1950 DATA 1,23,165,62,80,99,75,90,88,84,32 [2183]
,35
1960 DATA 1,22,192,99,100,85,99,69,77,88,3 [1456]
5,42
1970 DATA 1,28,171,66,84,86,90,94,90,89,30 [2293]
,38
1980 RESTORE 1900                          [922]
1990 INK 0,0:INK 1,15:MODE 2               [1269]
2000 nr=aktfig:GOSUB 2330                  [1285]
2010 LOCATE 6,23:PRINT"Ggst "CHR$(24)"h"CH [16262]
R$(24)"inlegen":LOCATE 24,23:PRINT"Ggst "C
HR$(24)"a"CHR$(24)"ufnehmen":LOCATE 43,23:
PRINT"Ggst "CHR$(24)"u"CHR$(24)"ebergeben"
:LOCATE 63,23:PRINT"Ggst "CHR$(24)"b"CHR$(
24)"enutzen"
2020 LOCATE 29,25:PRINT"Mit 'R' wieder zur [3396]
ueck!"
2030 a$=UPPER$(INKEY$):IF a$="R" THEN 2080 [1582]
2040 IF a$="U" THEN 2090                   [1279]
2050 IF a$="H" THEN 2400                   [1488]
```

```
2060 IF a$="A" THEN 2490                     [1176]
2070 GOTO 2030                               [365]
2080 MODE 0:PAPER 2:CLS:INK 0,18:INK 1,2:G [3983]
OSUB 470:GOTO 1860
2090 LOCATE 10,25:PRINT"Mit der Nummer ein [7648]
er Tasche ohne Inhalt kommen Sie zurueck!"
2100 LOCATE 1,23:PRINT"Geben Sie die Numme [7271]
r der Tasche ein, aus der Sie den Ggst ueb
ergeben wollen    ";:LOCATE 76,23
2110 INPUT tnr:IF tnr<1 OR tnr>6 THEN 2100 [2289]
2120 IF besitz(aktfig,tnr)=0 THEN PRINT CH [2071]
R$(7);:GOTO 1980
2130 adr=feld+xscr+x+(yscr+y)*xaus(ebene)  [2639]
2140 status(1)=PEEK(adr-1)-37:status(2)=PE [4491]
EK(adr+1)-37
2150 status(3)=PEEK(adr-xaus(ebene))-37    [2589]
2160 status(4)=PEEK(adr+xaus(ebene))-37    [3649]
2170 anz=4:MODE 2                          [1493]
2180 FOR l=1 TO 4                          [468]
2190 IF status(l)<1 OR status(l)>4 THEN an [4729]
z=anz-1:status(l)=255:GOTO 2210
2200 LOCATE 25,8+l*2:PRINT"Nr.   ";status( [4386]
l);" mit dem Namen ";name2$(status(l),1)
2210 NEXT                                  [350]
2220 IF anz=0 THEN 1980:'Keine Figur in de [2333]
r Naehe
2230 LOCATE 25,6:PRINT "In Ihrer Reichweit [8050]
e stehen nachfolgend":LOCATE 25,8:PRINT"au
fgefuehrte Personen:"
2240 LOCATE 8,25:PRINT"Mit der Eingabe ein [18370]
er nicht aufgefuehrten Zahl kommen Sie zur
ueck!":LOCATE 25,18:PRINT"Welcher Figur mo
echten Sie den":LOCATE 25,20:PRINT"Gegenst
and uebergeben (Nr.1-4)          ";:LOCATE 5
8,20
2250 INPUT spnr:IF spnr<1 OR spnr>4 THEN 2 [2384]
240
2260 IF spnr<>status(1) AND spnr<>status(2 [8098]
) AND spnr<>status(3) AND spnr<>status(4)
THEN PRINT CHR$(7):GOTO 1980:'rueckkehr
2270 MODE 2:INK 1,15,16:nr=spnr:GOSUB 2330 [3372]
2280 LOCATE 11,25:PRINT"Mit der Eingabe ei [10178]
ner Tasche mit Inhalt kommen sie zurueck!"
:LOCATE 6,23:PRINT"     In welche Tasche s
oll der Gegenstand gesteckt werden (Nr)
     ";:LOCATE 70,23
2290 INPUT tnr2:IF tnr2<1 OR tnr2>6 THEN 2 [2304]
280
2300 IF besitz(spnr,tnr2)<>0 THEN PRINT CH [784]
R$(7):GOTO 1980
2310 besitz(spnr,tnr2)=besitz(aktfig,tnr): [3342]
besitz(aktfig,tnr)=0
2320 GOTO 1980                             [367]
2330 LOCATE 1,1:PRINT "Name":LOCATE 21,1:P [9010]
RINT name2$(nr,1):LOCATE 40,1:PRINT"Geschl
echt":LOCATE 61,1:PRINT name2$(nr,2)
2340 FOR n=0 TO 1:FOR m=1 TO 6:READ a$:LOC [6080]
ATE n*39+1,1+2*m:PRINT a$:LOCATE n*39+21,1
+2*m:PRINT werte(nr,n*6+m):PRINT:NEXT:NEXT
2350 LOCATE 1,15:PRINT"Inhalt von Tasche:" [2726]
2360 LOCATE 1,17:PRINT"1)";gegenst$(besitz [4877]
(nr,1)):LOCATE 41,17:PRINT"2)";gegenst$(be
sitz(nr,2))
2370 LOCATE 1,19:PRINT"3)";gegenst$(besitz [4402]
(nr,3)):LOCATE 41,19:PRINT"4)";gegenst$(be
sitz(nr,4))
2380 LOCATE 1,21:PRINT"5)";gegenst$(besitz [7092]
(nr,5)):LOCATE 41,21:PRINT"6)";gegenst$(be
sitz(nr,6))
2390 RETURN                                [555]
2400 LOCATE 10,25:PRINT"Mit der Nummer ein [7648]
er Tasche ohne Inhalt kommen Sie zurueck!"
2410 LOCATE 1,23:PRINT"Geben Sie die Numme [8870]
r der Tasche ein, aus der Sie den Ggst hin
legen wollen      ";:LOCATE 75,23
2420 INPUT tnr:IF tnr<1 OR tnr>6 THEN 2410 [2450]
2430 IF besitz(aktfig,tnr)=0 THEN PRINT CH [2071]
R$(7);:GOTO 1980
2440 adr=feld+xscr+x+(yscr+y)*xaus(ebene)  [2639]
2450 IF PEEK(adr)=42 THEN PRINT CHR$(7):GO [2078]
TO 1980
2460 POKE tabelle,ebene:POKE tabelle+1,x+x [8615]
scr:POKE tabelle+2,y+yscr:POKE tabelle+3,b
esitz(aktfig,tnr):POKE tabelle+4,PEEK(adr)
2470 tabelle=tabelle+5:besitz(aktfig,tnr)= [4482]
0:POKE adr,42
2480 GOTO 1980                             [367]
2490 LOCATE 11,25:PRINT"Mit der Eingabe ei [11697]
ner Tasche mit Inhalt kommen Sie zurueck!"
:LOCATE 6,23:PRINT"     In welche Tasche s
oll der Gegenstand gesteckt werden (Nr)
     ";:LOCATE 70,23
2500 INPUT tnr2:IF tnr2<1 OR tnr2>6 THEN 2 [2270]
490
2510 IF besitz(aktfig,tnr2)<>0 THEN PRINT  [2966]
CHR$(7):GOTO 1980
2520 adr=feld+xscr+x+(yscr+y)*xaus(ebene)  [2639]
2530 IF PEEK(adr)<>42 THEN PRINT CHR$(7):G [3008]
OTO 1980
2540 FOR l=22000 TO tabelle STEP 5         [1064]
2550 IF PEEK(l)<>ebene THEN 2610           [1064]
2560 IF PEEK(l+1)<>x+xscr THEN 2610        [1065]
2570 IF PEEK(l+2)<>y+yscr THEN 2610        [1455]
2580 besitz(aktfig,tnr2)=PEEK(l+3)         [1342]
2590 POKE adr,PEEK(l+4)                    [325]
2600 adr=l:l=tabelle                       [1198]
2610 NEXT:IF adr>feld THEN STOP            [2358]
2620 FOR l=adr TO tabelle:POKE l,PEEK(l+5) [2663]
:NEXT
2630 tabelle=tabelle-5:GOTO 1980           [3151]
2640 CLS:LOCATE 1,10:PRINT name2$(aktfig,1 [2852]
);" sieht"
2650 IF PEEK(feld+x+xscr+(y+yscr)*xaus(ebe [3274]
ne))=42 THEN 2680
2660 PRINT:PRINT"nichts Besonderes."       [1630]
2670 GOTO 2760                             [387]
2680 FOR l=22000 TO tabelle STEP 5         [1064]
2690 IF PEEK(l)<>ebene THEN 2750           [1012]
2700 IF PEEK(l+1)<>x+xscr THEN 2750        [1013]
2710 IF PEEK(l+2)<>y+yscr THEN 2750        [1546]
2720 PRINT:PRINT"folgenden":PRINT:PRINT"Ge [3598]
genstand:"
2730 PRINT:PRINT gegenst$(PEEK(l+3))       [1730]
2740 l=tabelle                             [615]
2750 NEXT                                  [350]
2760 LOCATE 1,25:PRINT"Bitte Taste druecke [3021]
n";:CALL &BB18
2770 CLS:GOSUB 470:GOTO 1860               [1207]
```

```
10 MEMORY &9FFF                            [134]
11 FOR x=0 TO 544 STEP 2:READ b$:b1=VAL("& [5320]
"+LEFT$(b$,2)):b2=VAL("&"+RIGHT$(b$,2))
12 POKE &A000+x,b1:POKE &A000+x+1,b2:NEXT  [1838]
13 SAVE "creator4.bin",b,&A000,544         [1252]
14 DATA DD6E,00DD,6601,2271,A2C9,212D,0822 [2534]
15 DATA 0BA1,2150,C022,16A1,212D,0022,06A1 [1969]
16 DATA 21ED,B822,08A1,CDDE,A111,40C0,213B [2180]
17 DATA C00E,14CD,02A1,21A0,0022,48A1,DD6E [1356]
18 DATA 0222,4EA1,DD5E,04DD,5605,210F,C0C3 [2142]
19 DATA 29A1,21D3,0722,0BA1,2150,C022,16A1 [2515]
20 DATA 212D,0022,06A1,21ED,B022,08A1,CDDE [2570]
21 DATA A111,0FC0,2114,C00E,14CD,02A1,21A0 [1543]
22 DATA 0022,48A1,DD6E,0222,4EA1,DD5E,04DD [2095]
23 DATA 5605,213C,C0C3,29A1,21CE,F722,0BA1 [2181]
24 DATA 21B0,3F22,16A1,2132,0022,06A1,21ED [1432]
25 DATA B022,08A1,CDDE,A111,FFFD,215F,FD0E [2118]
26 DATA 12CD,02A1,2105,0022,48A1,2101,0022 [1598]
27 DATA 4EA1,DD5E,04DD,5605,210F,C0C3,29A1 [1844]
28 DATA 21CE,0722,0BA1,2150,C022,16A1,2132 [1223]
29 DATA 0022,06A1,21ED,B022,08A1,CDDE,A111 [2049]
30 DATA 0FC0,21AF,C00E,12CD,02A1,2105,0022 [1615]
31 DATA 48A1,2101,0022,4EA1,DD5E,04DD,5605 [1709]
32 DATA 21AF,C5C3,29A1,0608,C501,3200,EDB0 [2011]
33 DATA 01CE,07EB,09EB,09C1,10F0,C501,50C0 [2334]
34 DATA EB09,EB09,C10D,20E2,DD6E,00DD,6601 [1675]
35 DATA C3A4,A106,0AC5,D5E5,1A6F,2600,5F54 [2537]
36 DATA 2929,1929,2929,29ED,5B71,A219,EBE1 [2693]
37 DATA E5CD,62A1,E111,A000,19D1,EB01,8900 [1352]
38 DATA 09EB,C110,D6C9,DD5E,00DD,5601,DD6E [1744]
39 DATA 02DD,6603,0E10,0605,1A77,1323,10FA [1220]
40 DATA D511,FB07,1930,0411,50C0,19D1,0D20 [1674]
41 DATA E9C9,060A,21A0,0022,48A1,DD6E,0022 [1590]
42 DATA 4EA1,DD5E,02DD,5603,210F,C0D5,E5C5 [2052]
43 DATA CD29,A1C1,E111,0500,19D1,1310,F0C9 [1947]
44 DATA DD5E,08DD,5609,0E10,0605,C51A,E6AA [1917]
45 DATA 2003,7EE6,AA4F,1AE6,5520,037E,E655 [1951]
46 DATA B177,1323,C110,E77D,C6FB,6F3E,078C [2288]
47 DATA 6730,087D,C650,6F3E,C08C,670D,C2AC [1912]
48 DATA A1C9,0601,C5C5,DD6E,00DD,6601,DD7E [2242]
49 DATA 06E5,C32F,A1DD,5E00,DD56,01DD,6E02 [1327]
50 DATA DD66,03C3,AAA1,DD6E,00DD,6601,0E20 [1708]
51 DATA 060A,7EEE,FF77,2310,F911,F607,19D2 [2217]
52 DATA 1AA2,1150,C019,0DC2,06A2,C900,0000 [1454]
```



Na, logisch...

Basic-Kurs: Logische Ausdrücke und Boolesche Algebra

Ohne Logik wäre das gesamte Gebiet der Datenverarbeitung kaum vorstellbar. Einer der fundamentalen Ausdrücke eines logischen Zusammenhangs ist die einfache Bedingung. Auch das BASIC des CPC erlaubt es, die Ausführung einzelner Programmteilevon bestimmten Bedingungen abhängig zu machen. Knüpfen Sie mit uns ein Netz aus logischen Verbindungen. Aber verstricken Sie sich nicht!

Wenn es regnet, wird die Straße naß. Andrea oder Hans kommen heute zum Essen – vorausgesetzt, Horst kommt nicht.
Was das in einem BASIC-Kurs zu suchen hat? – Ganz einfach: Bei diesen scheinbar sinnlos zusammengestellten Sätzen handelt es sich um logische Ausdrücke, oder genauer: um bedingte Aussagen. Um den Einsatz bedingter Anweisungen geht es auch in dieser Folge unseres BASIC-Kurses.

Wenn das Wörtchen AND nicht wär'...

Der folgende Satz läßt sich auf einfache Weise "computergerecht" umformen:
"Die Glühbirne leuchtet, wenn Lichtschalter und Sicherung den Stromkreis schließen."
Der Satz nennt die Bedingungen, die erfüllt sein müssen, damit (als Ergebnis) die Glühbirne leuchtet. Anders geschrieben sieht das so aus:

```
Glühbirne_an = Lichtschalter_an
AND Sicherung_an
```

Das ist eine Gleichung – ebenso wie etwa "x = 5", nur daß die Variablen hier statt Zahlenwerten lediglich die Werte "wahr" oder "falsch" annehmen können. Das Rechnen mit solchen Werten nennt man Boolesche Algebra – nach dem Mathematiker Boole, der mit seiner binären Arithmetik die Verknüpfung logischer Zustände beschrieben hat.

Wahrheitsweite im Zahlengewand

Die logischen Zustände "wahr" und "falsch" werden in BASIC durch die Zahlenwerte -1 und 0 repräsentiert. Der Ausdruck "1 = 1" ist wahr, wird also durch -1 gekennzeichnet. Probieren Sie es aus:

```
PRINT (1 = 1)
```

Eine merkwürdige Art von Mathematik, nicht wahr? Aber immerhin: So lassen sich die Ergebnisse logischer Ausdrücke in "gewöhnliche" Formeln einbauen.
Was bedeutet nun das "AND" in der logischen Gleichung? Es besagt, daß der Ausdruck "A AND B" nur dann wahr ist, wenn A und B gleichzeitig wahr sind.
In diesem Zusammenhang ist es bei komplizierteren Gleichungen oft nützlich, eine sogenannte Wahrheitstabelle zu Rate zu ziehen. Diese stellt die einzelnen Bedingungen einer logischen Verknüpfung ("wenn...") als "Eingänge" und das Ergebnis ("dann...") als "Ausgang" dar. Für die **AND-Funktion** aus unserem Beispiel sieht die Wahrheitstabelle folgendermaßen aus:

Eingang1	Eingang2	Ausgang
wahr	wahr	wahr
falsch	wahr	falsch
wahr	falsch	falsch
falsch	falsch	falsch

Jeder möglichen Eingangskombination wird als Ergebnis der Wert "wahr" oder "falsch" zugeordnet. Außer der hier dargestellten AND-Verknüpfung gibt es noch einige andere logische Verknüpfungen. Wir wollen uns hier jedoch auf diejenigen beschränken, die vom BASIC des CPC unterstützt werden. Komplexere Verknüpfungen lassen sich dann daraus zusammensetzen.

Die drei logischen "Verkuppler"

Der altbekannte Blick ins Handbuch verrät, daß das CPC-BASIC drei logische Verknüpfungen kennt: AND, OR und XOR. Die noch fehlenden Wahrheitstabellen sehen folgendermaßen aus:

OR-Funktion:

Eingang1	Eingang2	Ausgang
wahr	wahr	wahr
falsch	wahr	wahr
wahr	falsch	wahr
falsch	falsch	falsch

```
10   REM Wischi
20   Heizung=0: REM 0=aus/-1=an
30   Motor=0: REM 0=aus/-1=an
40   Tuer=0: REM 0=offen/-1=zu
50   Startknopf=0: REM 0=aus/-1=an
60   INPUT "Startknopf(-1=an):",Startknopf
70   INPUT "Temperatur :",Temperatur
80   INPUT "Tuer :",Tuer
90   Motor = Startknopf AND Tuer
100  Heizung=(Temperatur<40) AND Tuer
110  PRINT "Heizung :",Heizung
120  PRINT "Motor :",Motor
130  GOTO 60
```

Das Programm "Wischi" löst das im Text beschriebene Waschmaschinen-Problem mit Hilfe logischer Ausdrücke

```
10   REM Waschi
20   INPUT"Startknopf(-1=an):",Startknopf
30   INPUT "Temperatur :",Temperatur
40   INPUT "Tuer :";Tuer
50   IF Startknopf=-1 THEN IF Tuer=-1 THEN Motor=-1
     ELSE Motor=0
60   IF Temperatur<40 THEN IF Tuer=0   THEN GOTO 90
70   Heizung=-1
80   GOTO 100
90   Heizung=0
100  PRINT "Heizung :",Heizung
110  PRINT "Motor :",Motor
120  GOTO 20
```

Das Programm "Waschi" arbeitet mit IF-THEN-Abfragen. Es ist leichter zu durchschauen, aber weniger elegant und langsamer als "Wischi"

XOR-Funktion:

Eingang1	Eingang2	Ausgang
wahr	wahr	falsch
falsch	wahr	wahr
wahr	falsch	wahr
falsch	falsch	falsch

Diese Funktionen lassen sich nun nicht nur auf jeweils zwei logische Variablen anwenden, sondern auch auf ganze Gruppen. Solche Gruppen bestehen, wenn man Zahlen zwischen 0 und 256 miteinander verknüpfen will, aus jeweils acht logischen Zuständen, nämlich Bits. Eine Gruppe von acht Bits ist bekanntlich ein Byte. Was passiert aber, wenn man eine logische Verknüpfung auf ein Byte anwendet?
Tippen Sie einmal folgendes ein:

```
PRINT 128 AND 64
```

Der CPC meldet 0. Wie kommt dieses Ergebnis zustande? Die Erklärung erfordert einen kleinen Besuch bei Herrn Boole – also einen Ausflug in die binäre Logik.
Eine Zahl besteht für unseren Rechner aus einer Reihe von Bits, die jedes für sich entweder "gesetzt" oder "nicht gesetzt" sind. Diese beiden Alternativzustände eines Bits kann man je nach Geschmack auch als "1" und "0" – oder eben als "wahr" und "falsch" bezeichnen.
Jedes Bit hat entsprechend seiner Position einen anderen Wert. Genau so kann ja auch im alltäglichen Dezimal-Zahlensystem die Ziffer 1 den Wert 1, 10 oder 10000 haben, je nachdem, an welcher Stelle sie steht. Die Bitwerte bilden ein Zweier-Zahlensystem. Man schreibt Binärzahlen der Bequemlichkeit halber in gleicher Richtung wie ihre dezimalen Kollegen, also von rechts nach links. Die Wertigkeit eines Bits erhält man somit durch stetes Verdoppeln der Wertigkeit seines rechten Nachbarn. Die Reihe lautet von rechts nach links 1, 2, 4, 8, 16, 32, ... bis 128, für das achte Bit eines Bytes. Für Zahlen, die mehr als ein Byte belegen, läßt sich die Reihe genau so weiterführen. Das niedrigstwertige Bit, nämlich das mit der Wertigkeit 1, steht immer ganz rechts, das höchstwertige Bit immer ganz links.
Wenn wir nun die Zahl 128 in Bits ausdrücken wollen, sähe das so aus:

```
10000000 = 128
```

Entsprechend die 64:

```
01000000 = 64
```

Wenn wir auf diese beiden Zahlen nun das logische AND anwenden, nehmen wir immer die Bits mit der jeweils gleichen Wertigkeit und verknüpfen sie:

```
     10000000
AND  01000000
=    00000000
```

Das mag ja ein ganz lustiger Effekt sein, aber wofür braucht man ihn?

Bitmanipulation auf Umwegen

Um einzelne Bits gezielt zu beeinflussen, gibt es in BASIC keine Befehle. Aber mit Hilfe der logischen Funktionen wird das möglich.
Die AND-Funktion ermöglicht es, einzelne Bits zu löschen. Die OR-Funktion setzt einzelne Bits, und mit der XOR-Funktion kann man die Zustände einzelner Bits umdrehen.
Angenommen, der CPC soll Punkte am Bildschirm setzen, deren Farbe ständig sämtliche Farbwerte von 0 bis 31 durchlaufen soll. Dies läßt sich entweder mit einer IF-THEN-Konstruktion erreichen oder mit einem AND-Befehl.

```
10 REM Punkte 1
20 Farbe=0
30 PLOT RND*640,RND*400,Farbe
40 Farbe=Farbe+1
50 IF Farbe=32 then Farbe=0
60 GOTO 30
```

Genauso kann man auch schreiben:

```
20 Farbe=0
30 PLOT RND*640,RND*400,Farbe
40 Farbe=(Farbe+1) AND 31
60 GOTO 30
```

Zeile 50 entfällt: Da alle höherwertigen Bits gelöscht werden, kann die Farbe nie größer als 31 werden. Bei diesem Trick ist jedoch Vorsicht geboten. Diese Begrenzung funktioniert nur bei Zahlen, die um eins kleiner sind als die Bit-Wertigkeiten, also 1, 3, 7, 15, bis 127 – bei anderen Zahlen haben manche Bits "zwischendrin" den Wert 0, und schon ist der Bereich nicht mehr lückenlos abgedeckt.
So weit, so gut. Sie sind jetzt in der Lage, logische Funktionen zu beschreiben. Deren Verhalten ist Ihnen durch die Benutzung der Wahrheitstabellen auch nicht mehr schleierhaft. Was bringen diese Funktionen konkret bei der Programmentwicklung? Sie erhöhen den Komfort beim Programmieren.

Zweimal große Wäsche

Die Programme "Wischi" und "Waschi", die in den Kästen zu diesem Artikel abgedruckt sind, sollen dies zeigen. Sie simulieren eine Waschmaschine. Der Motor soll nur dann laufen, wenn der Startknopf gedrückt und die Tür ge-

schlossen ist. Die Heizung soll hingegen beim Unterschreiten einer Temperatur von 40 Grad eingeschaltet werden, aber nur wenn die Tür geschlossen ist.
Vor allem an der Zahl der GOTOs merkt man deutlich den Unterschied zwischen "Wischi", das logische Ausdrücke verwendet, und "Waschi", das sich an die klassische IF-THEN-Abfrage hält.
Dabei kommt "Waschi" jedoch nur scheinbar ohne logische Verknüpfungen aus. Wir haben bisher klammheimlich so getan, als habe der IF-Befehl nichts mit Boolescher Algebra zu tun und ihn benutzt, ohne ihn genauer zu erklären.
In Wirklichkeit tun die Befehle IF-THEN-ELSE nichts anderes, als logische Ausdrücke auszuwerten. Wenn der Audruck nach dem IF wahr ist, dann wird der Teil nach dem THEN ausgeführt. Anderenfalls kommen die Befehle nach dem ELSE an die Reihe.

Ballspiele

Betrachten wir als Anwendung logischer Verknüpfungen zum Beispiel einen herumhüpfenden Ball.
Unser Ball besitzt eine Bewegungsrichtung. Sie besteht aus X-und Y-Bewegung.
Er soll sich geradlinig bewegen, bis er am Bildschirmrand anstößt. Wenn er links oder rechts anstößt, soll er seine X-Richtung umkehren. Bei einer Kollision oben oder unten wird entsprechend die Y-Richtung umgekehrt. Wie man sieht, steckt in diesem Beispiel eine Menge Logik. Wie schon beim Waschmaschinenbeispiel läßt sich auch diese Aufgabe auf unterschiedliche Art lösen. Zuerst die herkömmliche Lösung:
Wir setzen alle Variablen, wobei BX und BY die Position des Balls, VX und VY die Bewegung angeben. VX und VY sind entweder negative oder positive Zahlen, je nachdem, in welche Richtung sich der Ball bewegt.

```
10    REM Ball
20    BX=20:BY=10
30    VX=1:VY=1
```

Als nächstes wird der Ball gelöscht.

```
40    LOCATE BX,BY
50    PRINT " ";
```

Danach wird die neue Position berechnet.

```
60    BX=BX+VX
70    BY=BY+VY
```

Jetzt überprüft das Programm, ob der Ball am Bildschirmrand angestoßen ist.

```
80    IF BX<40 THEN 100
90    VX=-1
100   IF BX>1 THEN 120
110   VX=1
120   IF BY<25 THEN 140
130   VY=-1
140   IF BY>1 THEN 160
150   VY=1
```

Nachdem alle Werte korrigiert sind, können wir den Ball an seiner neuen Position ausgeben.

```
160   LOCATE BX,BY
170   PRINT "o";
```

Und das Ganze wieder von vorne:

```
180 GOTO 40
```

Sehr schnell und kompakt ist die Sache nicht. Dafür sind vor allem die IF-THEN-Abfragen und die Sprünge verantwortlich. Es geht aber auch anders.
Die Zustände "wahr" und "falsch" werden, wie eingangs bereits erwähnt, in BASIC durch die Zahlenwerte -1 und 0 repräsentiert. Mit diesen Schaltwerten, nämlich -1 für einen wahren und 0 für einen falschen Ausdruck, lassen sich unsere Bedingungen für die Ballbewegung in Gleichungen fassen. So sparen wir Platz und steigern gleichzeitig die Ausführungsgeschwindigkeit.
Der neue Abfrageteil lautet dann so:

```
60    VX=VX*(BX>39 OR BX<2)-
      VX*(BX<40 AND BX>1)
70    VY=VY*(BX>24 OR BY<2)-
      VY*(BY<25 AND BY>1)
80    BX=BX-VX*(BX<41 AND BX>0)
90    BY=BY-VY*(BY<26 AND BY>0)
```

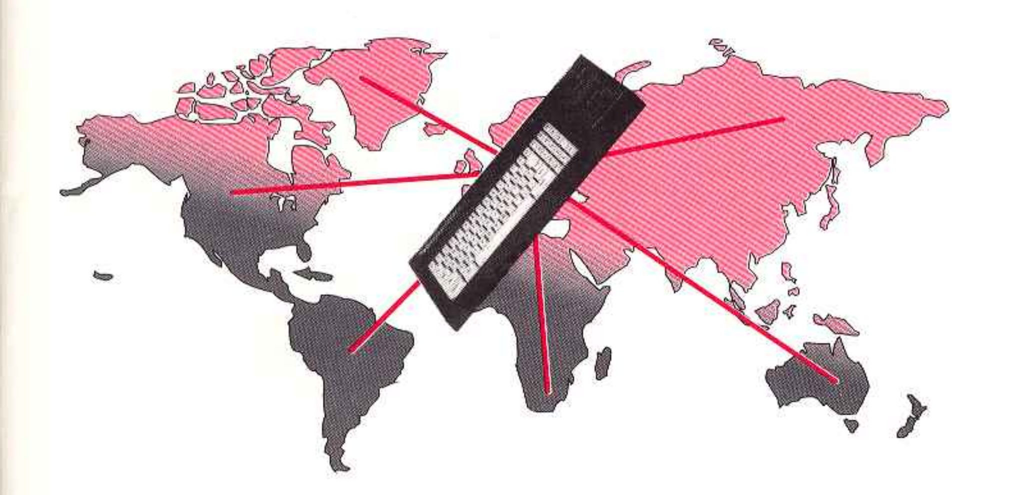
Logischer Ballbeschleuniger

Er ist wesentlich kompakter und schneller als die erste Version. Eine komplette Version des "Balls" finden Sie anschließend abgedruckt. In Zeile 190 beginnt die Initialisierung des Bildschirms und der Variablen. In bx und by sind die aktuellen Koordinaten gespeichert. Nach deren Änderung bleiben die alten Werte noch in ox und oy erhalten. Das ist nötig, um den Ball wieder (Zeile 320) löschen zu können. Die Variablen werden anschließend in Zeile 339 aktualisiert. Ab Zeile 250 findet man den Abfrageteil mit der logischen Auswertung. Die Version können Sie noch weiter ausbauen, indem Sie beispielsweise die Schwerkraft berücksichtigen, Hindernisse einbauen, einen Ballspieler hinterherlaufen lassen, und, und, und...
Wir wünschen Ihnen viel Spaß und gutes Gelingen.

Markus Kräutner/jg

```
100 REM ******************************     [1576]
110 REM **                          **     [409]
120 REM **          BALL.BAS        **     [476]
130 REM ** (c) 1992 Markus Kraeutner **    [2927]
140 REM **     & CPC International  **     [990]
150 REM **                          **     [409]
160 REM ******************************     [1576]
170 '                                      [117]
180 REM Bildschirm initialisieren          [3441]
190 MODE 1:INK 0,0:BORDER 6:INK 1,6        [2397]
200 '                                      [117]
210 REM Variablen initialisieren           [2187]
220 bx=20:by=20:ox=bx:oy=by                [1761]
230 vx=-1:vy=-1                            [1513]
240 '                                      [117]
```

```
250 vx=vx*(bx>39 OR bx<2)-vx*(bx<40 AND bx [2734]
>1)
260 vy=vy*(by>24 OR by<2)-vy*(by<25 AND by [3606]
>1)
270 bx=bx-vx*(bx<41 AND bx>0)              [2435]
280 by=by-vy*(by<26 AND by>0)              [1452]
290 '                                      [117]
300 REM Ball loeschen                      [1584]
310 LOCATE ox,oy                           [1234]
320 PRINT " ";                             [673]
330 ox=bx:oy=by                            [1779]
340 '                                      [117]
350 REM Ball ausgeben                      [920]
360 LOCATE bx,by                           [665]
370 PRINT "o";                             [638]
380 GOTO 250                               [419]
```



Verknotet mit der Außenwelt

Rechnernetze zum weltweiten Datenaustausch

Wer konsequent DFÜ betreibt und nicht nur Programme, sondern auch Briefe per Computer verschicken möchte, wird sicherlich irgendwann mit Begriffen wie Zerberus, Fido, Maus oder auch Live-Net konfrontiert werden. Was es damit auf sich hat? Wir wollen hier für Klärung sorgen.

Neben normalen Mailboxen im Stand-Alone-Betrieb gibt es auch ganze Gruppen von Mailboxen, die regelmäßig untereinander Daten austauschen. Bei diesen Daten handelt es sich größtenteils um Texte, die entweder für die Allgemeinheit oder aber für einen bestimmten User innerhalb dieses Netzes gedacht sind.

Eines der größten Netze, die Rede ist von Fido, entstand vor etwa 15 Jahren in den Vereinigten Staaten. Von Tom Jennings ins Leben gerufen, schlossen sich in den nachfolgenden Jahren weltweit tausende von interessierten Usern an dieses Netz an. Allein in Deutschland gibt es über 400 Nodes, die an Fido angeschlossen sind. Damit in diesem Netz jede einzelne Mailbox wiedergefunden werden kann, bekommt jede Box eine Nummer, im Fachjargon AKA genannt. Sie sieht etwa so aus:

z:nnn/bbb.pp

Wie Sie sehen, wird die Nummer in vier Bereiche unterteilt. Die Nummer läßt sich wie folgt entschlüsseln:

Z steht hierbei für Zone und umschreibt die Region, in welcher der Fido-Anschluß besteht. Die 2 an dieser Stelle bedeutet, daß die Box dem europäischen Raum zuzuordnen ist.

Die nachfolgende Zahl nnn gibt nun wieder Auskunft über das Land sowie das Netz, in welchem die Mailbox betrieben wird. Deutschland belegt hierbei die zwei Anfangsziffern 24 und 50.

Die Netznummer identifiziert nun Ihre Box ganz genau. Sie wird nur einmal verteilt und erlaubt es so jedem, eine Nachricht gezielt an Sie zu verschicken.

Eine letzte Unterteilung ist nun die Pointnummer. Und spätestens hier wird es für Sie interessant.

Eine eigene Mailbox mit dem CPC oder PCW zu betreiben, ist doch etwas utopisch. Zwar existieren für beide Rechner Pseudo-Mailboxprogramme, doch sind sie in Leistung und Komfort nicht das, was man sich wünscht. Eine Aufgabe für unsere Leser? Aber zurück zum Point. Die Pointnummer, die durch einen Punkt abgetrennt an eine Fido-Adresse angehängt wird, gestattet es Ihnen, auch ohne eigene Box im Fido-Netz aktiv zu werden.

So bieten viele Mailboxbetreiber ihren Usern gegen eine kleine Kostenbeteiligung an, über ihre Box Daten und Texte des Fido-Netzes zu empfangen und zu versenden.

Alles, was Sie nun benötigen, ist ein Pointprogramm, das speziell für Sie aufbereitete Fido-Daten automatisch empfangen kann. Als Beispiel sei hier das aus der englischen Public Domain stammende CRR0110 genannt.

Es macht im Grunde nichts anderes, als die für Sie auf der Stammbox versandfertig gepackte Datei automatisch anzufordern und zu übertragen. Gleichzeitig werden Ihre Texte an die entsprechenden Empfänger verschickt.

Damit der Stammrechner auch weiß, welche Daten für Sie interessant sind, müssen Sie ihm vorab einige Informationen über gewünschte Textbretter geben. Mehr hierüber erfahren Sie jedoch bei Ihrem Sysop.

Von Rechner zu Rechner

Wie wird man nun jedoch Point? Dies ist eigentlich gar nicht so schwierig. Am besten, man fragt zunächst den Sysop seiner Stammbox, in der man bisher seine Nachrichten und Programme bezogen hat.

Ist dieser an einen Netzverbund angeschlossen, hat dieser Zugriff auf eine Liste, in welcher alle weiteren Boxen mit Netzzugriff aufgelistet sind. So kann er Ihnen dann mit Sicherheit eine Box in Ihrer näheren Umgebung nennen, bei welcher Sie dann genaueres erfahren können.

Netze ohne Anschluß?

Im Laufe der Zeit haben aber auch noch andere Netze Fuß gefaßt. Fido ist also bei weitem nicht das einzige. Die prominentesten Netze in Deutschland sind unter anderen Zerberus, Maus und Live-Net, die jeweils mit bestimmten Themenbereichen aufwarten.

Da mancher vielleicht an einzelnen Themenbereichen mehrerer Netze interessiert ist, gibt es noch sogenannte Gateways. Hier handelt es sich um Knotenpunkte, auf denen Daten mehrerer Netze ineinanderlaufen und miteinander vermischt werden.

rs

AKA – Benutzernummer, die einen User weltweit identifiziert
Node – Eine Mailbox, die sich in einem Netzverbund befindet
Point – Benutzer, der regelmäßig speziell für ihn zusammengesuchte Texte/Dateien, vom Sysop seiner Mailbox bekommt
Sysop – Betreiber einer Mailbox
Textbretter – In den einzelnen Netzverbänden werden Nachrichten in unterschiedlichen Textbrettern abgelegt. Als Beispiele seien hier nur Politik, Recht, Musik, Film und auch Sex/Erotik genannt.

Sanfte Schiebung

Assembler-Ecke: Phasen-Animation und wandernde Sprites

Bewegliche, vom Bildschirmhintergrund unabhängige Figuren sind eine wichtige Voraussetzung, wenn es darum geht, ein flottes Spiel auf dem CPC zu programmieren. Dieser Beitrag gibt Ihnen ein bequemes Werkzeug zur Verarbeitung animierter Sprites an die Hand.

Im letzten Heft konnten Sie bereits einiges über den Bildschirmspeicher des CPC und über die Programmierung von Sprites lesen. Was wir dort bereits in der Theorie erörtert haben, wird nun praktisch. Das Programm, das Sie im Listingkasten zu diesem Artikel finden, schiebt ein mit dem Joystick kontrolliertes Sprite weich über den Bildschirm und täuscht mit Hilfe mehrerer Bildphasen eine Bewegung vor.

Von der Theorie zur Praxis

Unser Sprite kennt zwei "Blickrichtungen", die mit der Joystickbewegung umgeschaltet werden: rechts und links. Wir haben die Darstellung des Sprites in drei Phasen pro Blickrichtung aufgeteilt; es gibt also insgesamt sechs Phasen. Die Bitmaps für die einzelnen Phasen werden gemeinsam in dem Binärfile "Sprite.Spr" gespeichert, das der abgedruckte Basic-Lader erzeugt.

Das Neue an dem abgedruckten Assemblerprogramm ist die Auswahl-Logik für die gerade aktive Phase. Über eine Parameterliste läßt sich die Spritedarstellung steuern. Ein Teil des Programms wird in die Interruptschleife eingehängt und so immer wieder aufgerufen. Dabei werden Parameter wie die momentane Bildschirmadresse, Phasennummer und Phasenwechselgeschwindigkeit aus der Liste ausgelesen. Die Verknüpfung dieser Parameter erlaubt die Spritedarstellung.

Die Verschiebung des Sprites auf dem Bildschirm erfolgt in den Routinen, die bei der Joystickabfrage angesprungen werden. In diesen Routinen wird jedoch nur ein Parameter aus der Liste, nämlich die Bildschirmadresse, geändert. Der Aufbau des Assemblerprogramms ist dabei wie folgt:

joy:	die Joystickabfrage und die jeweilige Verzweigung
klickon:	Hiermit wird die Routine "show" in die Interruptkette eingehängt (bei jedem Bildrücklaufstrahl soll "show" aufgerufen werden).
klickof:	hängt die Routine wieder aus.
copy:	sichert den Hintergrund.
write:	stellt das Sprite dar.
write2	stellt den Hintergrund wieder her.
oben:	Abhängig davon, ob das Sprite den oberen Rand erreicht hat, wird die Bildschirmadresse geändert. Dies wird durch Abfragen logischer Koordinaten festgestellt. Die Änderung der Adresse erfolgt nur, wenn bereits eine vorherige Änderung registriert wurde.
unten, links und *rechts:*	siehe oben
show:	In dieser Interruptroutine wird das Sprite dargestellt.

Die Grafikdaten der einzelnen Sprite-Bewegungsphasen werden an die Adressen SPR1-SPR6 geladen, wobei die ersten drei die Daten für die Blickrichtung links aufnehmen. In dem "sprani" genannten Teil der Liste stehen die Adressen der Phasen, die aufeinander folgen sollen. Aus dieser Liste wird die jeweilige Adresse der Bewegungsphase ausgelesen.

Die Liste wird von oben nach unten durchlaufen, anschließend geht es wieder oben los. Am Beginn der Liste "tabell" steht die aktuelle Bildschirmadresse. Es folgt die Adresse mit dem Zeiger auf die Liste "sprani", die die Bewegungsphasenadressen enthält. Das nächste Listenelement ist der Wert für die aktuelle Phase (0-3), gefolgt vom momentanen Wert für die Animationsgeschwindigkeit – wenn dieser Null ist, wird eine Phase weitergeschaltet. Das Schlußlicht bildet der dazugehörige Nachladewert.

In der Liste "tabel2" steht zuerst die Bildschirmadresse der alten Position, dahinter dann die Adresse der gesicherten Hintergrunddaten.

In der Routine "phase" wird in Abhängigkeit von einigen Parametern der Zeiger auf die aktuelle Darstellungsphase gebildet. Genaueres dazu können Sie dem Listing entnehmen.

Durch die Verwendung der einzelnen Listen ist es möglich, mehrere Sprites zu verwalten und zu animieren. Sie müssen sich jedoch streng an den Aufbau der jeweiligen Listen halten, wenn Sie sie erweitern.

Abschließend noch ein Tip: Wenn Sie die Anzahl der Darstellungsphasen erhöhen wollen, müssen Sie die Liste "sprani" entsprechend erweitern und in der Routine "phase" den Wert hinter dem Befehl "and" ändern. Für acht verschiedene Phasen muß dieser Wert beispielsweise 7 betragen.

Das abgedruckte Listing entspricht bis auf die weggelassenen Zeilennummern und REM-Zeichen der Schreibweise unseres "CPC-Assemblers". Für einige andere Assembler (etwa "Devpac") müssen die Additionszeichen "*" in Pluszeichen umgewandelt werden.

Matthias Fink/Ulrich Schmitz/sz

```
; Sprites.Asm
; erzeugt "Sprites.Bin"
; (c) 1992 Matthias Fink & CPC International
;
        org   &a000
        ld    hl,(tabell)
                          ;Sichern des Hintergrunds vor
        ld    de,puffer   ;dem ersten Darstellen
        call  copy
        call  kickon
joy     call  &bb24       ;Joystickabfrage
        ld    a,h
        ld    c,a
        bit   0,c
        call  nz,oben
        bit   1,c
        call  nz,unten
        bit   2,c
        call  nz,links
        bit   3,c
        call  nz,rechts
```

```
        bit   4,c
        jp    nz,kickof
        jp    joy
kickon  ld    hl,event    ;Einbaengen in Interruptschleife
        ld    b,&81
        ld    c,0
        ld    de,show
        call  &bcef
        ld    hl,takt
        call  &bcda
        ret
kickof  di                ;Aushaengen
        ld    hl,takt
        call  &bcdd
        ret
copy    ld    c,16        ;siehe CPC 8/9'92, S. 47-48
copy1   ld    b,5
copy2   ld    a,(hl)
        ld    (de),a
        inc   hl
        inc   de
```

```
        djnz  copy2
        ld    a,l
        add   a,&fb
        ld    l,a
        ld    a,7
        adc   a,h
        ld    h,a
        jr    nc,copy4
        ld    a,l
        add   a,&50
        ld    l,a
        ld    a,&c0
        adc   a,h
        ld    h,a
copy4   dec   c
        jp    nz,copy3
        ret
write   ld    c,16        ;siehe oben
writ3   ld    b,5
writ2   push  bc
        ld    a,(de)
        and   &aa
        jr    nz,halb1
        ld    a,(hl)
        and   &aa
halb1   ld    c,a
        ld    a,(de)
        and   &55
        jr    nz,halb2
        ld    a,(hl)
        and   &55
halb2   or    c
        ld    (hl),a
        inc   de
        inc   hl
        pop   bc
        djnz  writ2
        ld    a,l
        add   a,&fb
        ld    l,a
        ld    a,7
        adc   a,h
        ld    h,a
        jr    nc,writ4
        ld    a,l
        add   a,&50
        ld    l,a
        ld    a,&c0
        adc   a,h
        ld    h,a
writ4   dec   c
        jp    nz,writ3
        ret
write2  ld    c,16        ;siehe oben
schre2  ld    b,5
schre3  ld    a,(de)
        ld    (hl),a
        inc   de
        inc   hl
        djnz  schre3
        ld    a,l
        add   a,&fb
        ld    l,a
        ld    a,7
        adc   a,h
        ld    h,a
        jr    nc,schre4
        ld    a,l
        add   a,&50
        ld    l,a
        ld    a,&c0
        adc   a,h
        ld    h,a
schre4  dec   c
        jp    nz,schre2
        ret
oben    ld    hl,yalt     ;Nur wenn yalt und ypos uebereIn-
        ld    a,(ypos)    ;stimmen (also eine vorherige Po-
        xor   (hl)        ;sitionsaenderung schon "bemerkt"
        ret   nz          ;wurde), erfolgt eine erneute
                          ;Aenderung
        ld    a,(ypos)    ;Abfrage, ob Rand erreicht ist
        cp    184
        ret   z
        add   a,2
        ld    (ypos),a    ;Aenderung der Koordinaten
        ld    a,(tabell+1)
                          ;Aend. der Bildschirmadresse
        sub   16          ;in der Liste "tabell"
        ld    (tabell+1),a
        bit   6,a
        ret   nz
        ld    hl,(tabell)
        ld    de,&3fb0
        add   hl,de
        ld    (tabell),hl
        ret
unten   ld    hl,yalt     ;siehe oben
        ld    a,(ypos)
        xor   (hl)
        ret   nz
        ld    a,(ypos)
        cp    0
        ret   z
        sub   2
        ld    (ypos),a
        ld    a,(tabell+1)
        add   a,16
        ld    (tabell+1),a
        ret   nc
        ld    hl,(tabell)
        ld    de,&c050
        add   hl,de
        ld    (tabell),hl
        ret
links   ld    hl,xalt     ;siehe oben
        ld    a,(xpos)
        xor   (hl)
        ret   nz
        ld    a,0
        ld    (vers),a
        ld    a,(xpos)
        cp    0
        ret   z
        dec   a
        ld    (xpos),a
        ld    hl,(tabell)
        dec   hl
        ld    (tabell),hl
        ret
rechts  ld    hl,xalt     ;siehe oben
        ld    a,(xpos)
        xor   (hl)
        ret   nz
        ld    a,1
        ld    (vers),a
        ld    a,(xpos)
        cp    75
        ret   z
        inc   a
        ld    (xpos),a
        ld    hl,(tabell)
        inc   hl
        ld    (tabell),hl
        ret
show    di
        push  af          ;Sichern der Register
        push  de
        push  bc
        push  hl
        push  ix
        ld    ix,tabell   ;Anfang der Parameterdaten
        ld    hl,(tabel2)
                          ;Adresse der alten Bildposition
        ld    de,(tabel2+2)
                          ;Adresse des Hintergrundpuffers
        call  write2      ;Herstellen des Hintergrunds
        ld    hl,(tabell)
                          ;akt. Bildschirmadresse
        ld    (tabel2),hl
        ld    de,(tabel2+2)
                          ;Pufferadresse
        call  copy        ;Sichern des Hintergrunds
        ld    hl,(tabell+2)
                          ;Anfangsadresse der Phasenliste
        ld    c,(ix+4)    ;akt. Phasennummer (0-3)
        ld    b,(ix+5)    ;momentaner Verzoegerungswert
        ld    e,(ix+6)    ;dazugehoeriger Nachladewert
        ld    a,(vers)    ;Version: 0=links 1=rechts
        call  phase       ;Ermitteln der Grafikadresse
        ld    (ix+5),b
        ld    (ix+4),a
        ld    hl,(tabell)
        call  write       ;Darstellen der akt. Phase
        ld    a,(xpos)    ;Aktualisieren d. log. Koord.
        ld    (xalt),a
        ld    a,(ypos)
        ld    (yalt),a
        pop   ix
        pop   hl
        pop   bc
        pop   de
        pop   af
        ei
        ret
phase   sla   a           ;In a: 0 fuer links, 1 fuer rechts
        sla   a           ;mal 8, weil 4 Grafikadressen
                          ;pro Richtung
        sla   a           ;in der Liste "sprani"
        add   a,c         ;Dazu die aktuelle Phase doppelt,
        add   a,c         ;da eine Adresse 2 Bytes enthält
        add   a,l         ;plus Anfangsadr. d. Liste "sprani"
        ld    l,a
        ld    a,h
        adc   a,0
        ld    h,a         ;In hl steht Adresse. d. akt. Phase
        ld    a,c
        djnz  phse2       ;Herunterzaehlen des Schleifenwerts
        ld    b,e         ;wenn Null, dann nachladen
        ld    a,c         ;Erhoehen der Phasennummer
        inc   a           ;(bzw. des Pointers)
        and   &03         ;und Beschraenkung auf Werte
                          ;zwischen 0 und 3,
phse2                     ;da in der Liste "sprani" nur
        ld    e,(hl)      ;4 Phasen pro Richtung
        inc   hl          ;generiert sind
        ld    d,(hl)
        ret
tabell  defw  &c000       ;Bildschirmadresse d. Sprites
        defw  sprani      ;Anfangsadresse der Phasenliste
pha     defb  0           ;Nummer der akt. Phase
        defb  4,4         ;Phasenwechselverzoegerung,
                          ;Nachladewert
tabel2  defw  &c000       ;Bildschirmadr. d. alten Position
        defw  puffer      ;Adresse fuer die Hintergrunddaten
sprani  defw  spr1        ;Adresse fuer Phase 1 links
        defw  spr2        ;Adresse fuer Phase 2 links
        defw  spr3        ;Adresse fuer Phase 3 links
        defw  spr2        ;Adresse fuer phase 2 links
spran2  defw  spr4        ;Adresse fuer Phase 1 rechts
        defw  spr5        ;Adresse fuer Phase 2 rechts
        defw  spr6        ;Adresse fuer Phase 3 rechts
        defw  spr5        ;Adresse fuer Phase 2 rechts
puffer  defs  80          ;Puffer fuer den Hintergrund
xpos    defb  0           ;logische Koordinaten
xalt    defb  0
ypos    defb  184
yalt    defb  184
vers    defb  1           ;0 = links, 1 = rechts
takt    defs  2
event   defs  6
spr1    defs  80          ;Bereich fuer die Grafikdaten
spr2    defs  80          ;der einzelnen Phasen
spr3    defs  80
spr4    defs  80
spr5    defs  80
spr6    defs  80          ;Ende des Listings
```

```
100 MEMORY &9FFF                                [134]
110 FOR adr=0 TO &1E0                           [765]
120 READ b$:b=VAL(b$)                           [858]
130 POKE &A000+adr,b                            [611]
140 NEXT                                        [350]
150 SAVE"sprites.spr",b,&A000,&1E0              [2307]
160  DATA 3,3,2,0,0,3,0,0,0,0                   [874]
170  DATA 3,0,0,0,0,2,2,0,0,0                   [575]
180  DATA 2,2,0,0,0,2,1,0,0,0                   [813]
190  DATA 2,1,0,0,0,2,0,2,0,0                   [897]
200  DATA 0,0,2,0,0,0,0,1,0,0                   [1217]
210  DATA 0,0,1,0,0,0,0,0,2,0                   [1237]
220  DATA 0,0,0,2,0,0,0,0,0,0                   [1108]
230  DATA 0,0,0,0,0,0,0,0,0,0                   [1269]
240  DATA 0,0,0,0,0,0,0,0,0,0                   [1269]
250  DATA 0,1,0,0,0,0,2,0,0,0                   [826]
260  DATA 0,2,0,0,0,1,0,0,0,0                   [953]
270  DATA 1,0,0,0,0,3,3,3,2,0                   [1069]
280  DATA 3,3,3,2,0,1,0,0,0,0                   [913]
290  DATA 1,0,0,0,0,0,2,0,0,0                   [962]
300  DATA 0,2,0,0,0,0,1,0,0,0                   [1148]
310  DATA 0,0,0,0,0,0,0,0,0,0                   [1269]
320  DATA 0,0,0,0,0,0,0,0,0,0                   [1269]
```

```
330  DATA 0,0,0,0,0,0,0,0,2,0                   [1325]
340  DATA 0,0,0,2,0,0,0,1,0,0                   [1124]
350  DATA 0,0,1,0,0,0,0,2,0,0                   [1086]
360  DATA 2,0,2,0,0,2,1,0,0,0                   [1010]
370  DATA 2,1,0,0,0,2,2,0,0,0                   [805]
380  DATA 2,2,0,0,0,3,0,0,0,0                   [828]
390  DATA 3,0,0,0,0,3,3,2,0,0                   [783]
400  DATA 0,0,1,3,3,0,0,0,0,3                   [1148]
410  DATA 0,0,0,0,3,0,0,0,1,1                   [1396]
420  DATA 0,0,0,1,1,0,0,0,2,1                   [1082]
430  DATA 0,0,0,2,1,0,0,1,0,1                   [1465]
440  DATA 0,0,1,0,0,0,0,2,0,0                   [1086]
450  DATA 0,0,2,0,0,0,1,0,0,0                   [1078]
460  DATA 0,1,0,0,0,0,0,0,0,0                   [439]
470  DATA 0,0,0,0,0,0,0,0,0,0                   [1269]
480  DATA 0,0,0,0,0,0,0,0,0,0                   [1269]
490  DATA 0,0,0,2,0,0,0,0,1,0                   [1104]
500  DATA 0,0,0,1,0,0,0,0,0,2                   [1694]
510  DATA 0,0,0,0,2,0,1,3,3,3                   [1373]
520  DATA 0,1,3,3,3,0,0,0,0,2                   [1136]
530  DATA 0,0,0,0,2,0,0,0,1,0                   [1210]
540  DATA 0,0,0,1,0,0,0,0,2,0                   [1680]
```

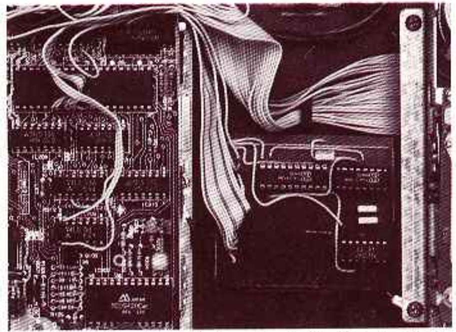
```
550  DATA 0,0,0,0,0,0,0,0,0,0                   [1269]
560  DATA 0,0,0,0,0,0,0,0,0,0                   [1269]
570  DATA 0,0,0,0,0,0,1,0,0,0                   [1201]
580  DATA 0,1,0,0,0,0,0,2,0,0                   [786]
590  DATA 0,0,2,0,0,0,0,1,0,0                   [1217]
600  DATA 0,0,1,0,1,0,0,0,2,1                   [912]
610  DATA 0,0,0,2,1,0,0,0,1,1                   [1655]
620  DATA 0,0,0,1,1,0,0,0,0,3                   [1320]
630  DATA 0,0,0,0,3,0,0,1,3,3                   [1377]
640  DATA 0,0,0,0,0,0,0,0,0,0                   [1269]
```

```
10 ' Startprogramm SPRITES.BAS                  [1134]
20 INK 0,0:BORDER 0:MODE 0                      [1295]
30 INK 1,3:INK 8,24:MEMORY &9FFF                [1783]
40 LOAD"sprites.bin",&A000                      [1932]
50 LOAD"sprites.spr",&A233                      [1719]
60 FOR n=1 TO 499:PRINT CHR$(INT(RND*220)+      [1999]
35);:NEXT
70 CALL &A000                                   [637]
```

Booten vom EPROM

CP/M ohne Ladezeiten

Der CPC als Z80-Computer ist prädestiniert dafür, mit dem Betriebssystem CP/M benutzt zu werden. Wie Sie nun auch ohne Bootdiskette auskommen, zeigt Ihnen dieser Beitrag.



Die Erweiterung wird unterhalb des Zahlenblocks eingebaut

Viele Nutzer des CPC arbeiten oft unter CP/M. Ist man gezwungen, häufig zwischen CP/M und BASIC zu wechseln, stört die Ladezeit erheblich. Außerdem muß man ständig eine CP/M-Systemdiskette zur Hand haben oder auf der Arbeitsdiskette fast 40 Kilobyte für das Betriebssystem reservieren. Ist das wirklich nötig? Kann man nicht das CP/M ebenso wie das AMSDOS auf Knopfdruck verfügbar machen?
Es ist möglich. Die hier vorgestellte Lösung wurde speziell für den CPC 6128 ausgearbeitet. Zur Realisierung dieses Eigenbaus ist leider ein kleiner Eingriff in den CPC nötig.

Kurz und schmerzlos

Die im folgenden beschriebenen Arbeiten sollten Sie äußerst vorsichtig durchführen. Für eventuell entstehende Schäden können wir nämlich keine Haftung übernehmen.

Sie beginnen mit dem Aufbau der Platine. Zuerst löten Sie die Sockel auf und anschließend die 100 nF-Kondensatoren. Diese dienen zum Abblocken der Betriebsspannung gegen Störungen. Nun noch die Diode und die Brücken a) bis j), und die Leiterplatte ist fertig.
Die Fassung fürs EPROM wird jetzt Pin für Pin "huckepack" aufs BASIC-ROM(IC103) gelötet. Die Pins 22 und 27 werden dabei frei gelassen und leicht weggebogen, so daß sie nicht die Anschlüsse des ROM berühren. Die Lötzeit sollte so kurz wie möglich gehalten werden. Anschließend versorgen Sie die Platine nach der Verbindungsliste mit den notwendigen Signalen. Vergessen Sie auch nicht die am EPROM-Sockel freigelassenen Verbindungen Pin 22 und Pin 27. Wünschen Sie das EPROM abschaltbar, so verbinden Sie den Eingang von IC3B Pin6 über einen Umschalter mit IC2 Pin 9 oder +5 Volt. Alle Kabel sollten so kurz wie möglich gehalten werden. Jetzt brauchen nur noch das EPROM und die anderen ICs in die Fassungen eingesetzt zu werden. Die fertige Leiterplatte kann bequem zwischen der Hauptplatine und dem Diskettenlaufwerk untergebracht werden.

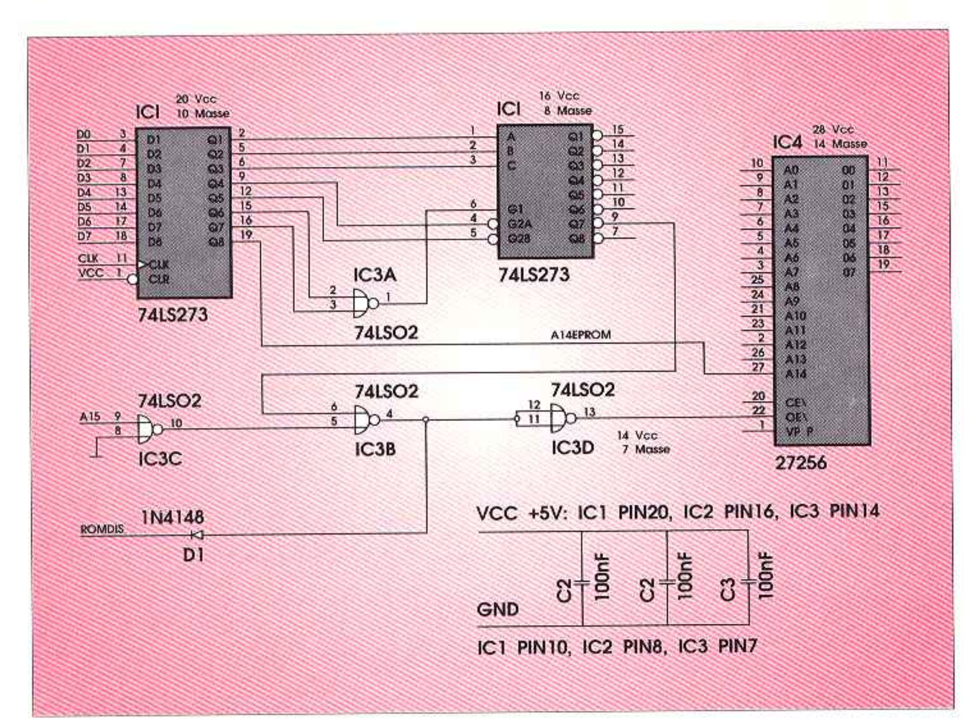


Stückliste:

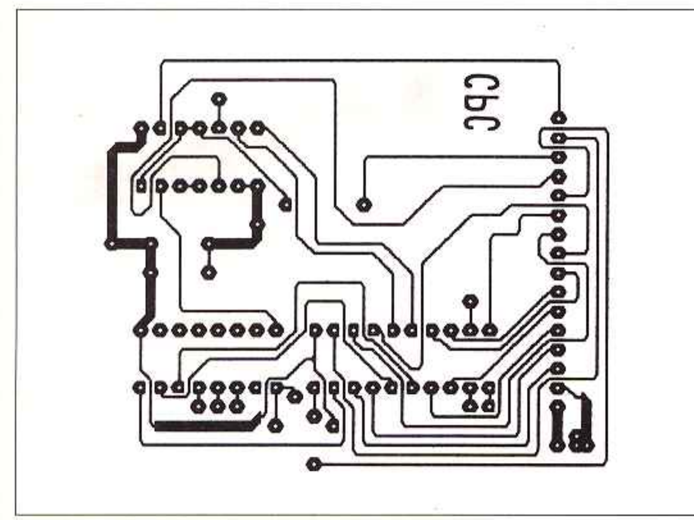
IC1	74LS273
IC2	74LS138
IC3	74LS02
IC4	27256-15
3	Kondensatoren 100nF keramisch
1	Diode 1N4148
1	Sockel 28polig
1	Sockel 20polig
1	Sockel 16polig
1	Sockel 14polig
	Leiterplatte gemäß Muster
	Kabel

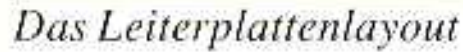
Für „Selbstbrenner": Das Programm ROM.BAS liest von der lCP/M -Systemdiskette die Datei C10CPM3.EMS ein und speichert sie in zwei Binärdateien ab. Diese werden dann in ein 32-kByte-EPROM gebrannt. Im ersten Teil des CPM-EPROMs (CPM1.ROM) befindet sich dann auch der Programmteil, der das ROM in das Betriebssystem einbindet, indem er den RSX-Befehl lCPM3 zur Verfügung stellt. Wie dieser Befehl vermuten läßt, startet er das CPM Plus aus dem EPROM. Anschließend übernimmt dieses die Kontrolle und versucht, die Datei PROFILE.SUB auf der Diskette zu finden.
Die Schaltung ist für den CPC 6128 in Standardausrüstung vorgesehen. Sollten Sie Hardwareerweiterungen benutzen, und die Schaltung arbeitet nicht mit diesen zusammen, so lesen Sie bitte die Infodatei auf der Databox.

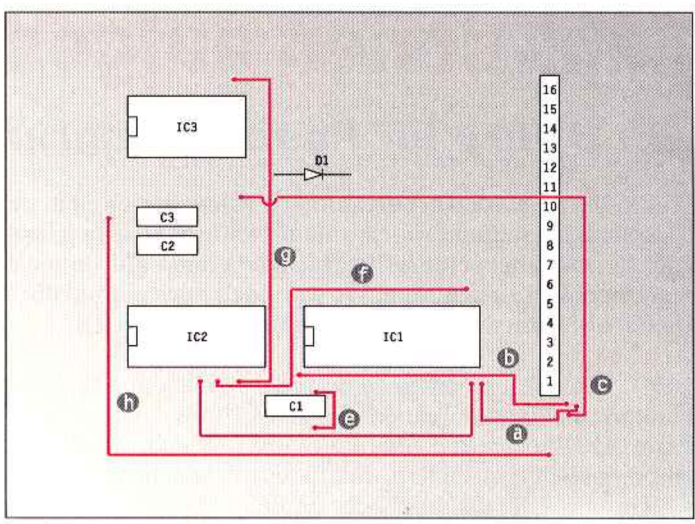
Michael Kruse/jg



Der Schaltplan der Leiterplatte




Das Leiterplattenlayout



Der Bestückungsplan

Verbindungsliste:

Signal	findet man an
D0	IC 201 PIN 6
D1	IC 201 PIN 7
D2	IC 201 PIN 8
D3	IC 201 PIN 9
D4	IC 201 PIN 10
D5	IC 201 PIN 11
D6	IC 201 PIN 12
D7	IC 201 PIN 13
A15	IC 205 PIN 1
CLK	IC 212 PIN 3
ROMDIS	der Kathode der Diode D201 (oberhalb von IC 203; an der Seite mit dem Ring anlöten)
+5V, VCC	am Ein/Aus-Schalter
GND	breiten Kupferbahnen, auf denen das Abschirmblech gelötet ist

So fertigen Sie die Leiterplatte an:
Sie beginnen damit, das Layout in eine Form zu bringen, in der es möglich ist, es auf fotografischem Weg weiterzubearbeiten. Die beste Variante ist, das Ganze auf Folie zu bringen. Hier kann man fotografieren oder reprografieren. Auch mit einigen Fotokopierern ist das möglich. Eine einfache Möglichkeit ist auch, das Layout auf Transparentpapier oder Folie mit einem passenden Stift nachzuzeichnen. Mit dieser Vorlage wenden Sie sich dann an einen Elektronik-Händler in Ihrer Nähe. Der kann Ihnen dann möglicherweise weiterhelfen, eine fotografisch beschichtete Leiterplatte zu belichten und zu ätzen. Auch in vielen Bastlerzeitschriften wird per Annonce ein Leiterplattenservice angeboten. Nur noch bohren, und die Leiterplatte ist fertig.

Unser Angebot:
Falls Sie keinen EPROM-Brenner besitzen und auch kein Elektronik-Händler in der Nähe ist, der Ihnen das benötigte EPROM brennt: Schicken Sie einen 10-DM-Schein im Brief zusammen mit einem **frankierten** Rückumschlag an die Redaktion. Wir senden Ihnen dann ein bereits programmiertes EPROM 27256.
(Falls Sie das EPROM mit veränderter ROM-Nummer gebrannt haben möchten, vergessen Sie bitte nicht, uns das mitzuteilen.)
Unsere Adresse:
Redaktion CPC International
Stichwort: CPC-EPROM
Fuldaer Str. 6
3440 Eschwege

```
100 'ROM.BAS                                   [976]
110 'CPC 6128                                  [593]
120 'Dieses Programm erzeugt zwei Dateien     [4586]
  fuer eine CP/M 3.0 EPROM-Version
130 'Auf der Diskette muss sich die Datei     [3374]
  C10CPM3.EMS befinden
140 '(c)1992 Michael Kruse                    [1651]
150 '   & CPC International                   [2118]
160 '                                          [117]
170 MEMORY &1FFF                               [710]
180 zeile=170:adr=&A100                        [595]
190 check=0                                    [177]
200 FOR i=1 TO 8                               [462]
210 READ a$:IF a$="END" THEN 280              [1350]
220 a=VAL("&"+a$):POKE adr,a                  [1324]
230 adr=adr+1:check=check+a                   [1154]
240 NEXT                                       [350]
250 READ a$:summe=VAL("&"+a$):IF summe<>ch    [4824]
eck THEN PRINT CHR$(7)"Fehler in Zeile"zei
le:END
260 zeile=zeile+10                             [568]
270 GOTO 190                                   [407]
280 '                                          [117]
290 CALL &A100                                 [633]
300 SAVE"CPM1.ROM",B,&2000,&3C62               [699]
```

```
310 SAVE"CPM2.ROM",B,&5C62,&2800              [2320]
320 END                                        [110]
330 DATA CD,65,BC,21,35,A1,06,0B,02F6          [955]
340 DATA 11,00,98,CD,77,BC,D0,21,039A          [939]
350 DATA 40,A1,11,00,20,01,62,00,0175         [1314]
360 DATA ED,B0,EB,01,00,64,E5,C5,0497         [1706]
370 DATA CD,80,BC,C1,E1,77,23,0B,0450         [1198]
380 DATA 78,B1,C2,1E,A1,00,11,00,02BB         [1597]
390 DATA 98,CD,7A,BC,C9,43,31,30,0408         [1776]
400 DATA 43,50,4D,33,2E,45,4D,53,0226         [1138]
410 DATA 01,00,00,00,0C,C0,C3,19,01A9         [2152]
420 DATA C0,C3,1A,C0,52,4F,4D,43,038E         [1601]
430 DATA 50,4D,33,A0,43,50,4D,B3,0303         [1771]
440 DATA 00,C9,21,37,C0,7E,B7,28,033E         [1125]
450 DATA 08,E5,CD,5A,BB,E1,23,18,03EB         [1658]
460 DATA F4,21,52,C0,11,F0,0B,01,0334         [1569]
470 DATA 10,3C,ED,B0,C3,F0,0B,0C,03B3         [1357]
480 DATA 52,4F,4D,2D,56,65,72,73,02BB         [1215]
490 DATA 69,6F,6E,20,56,31,20,62,026F         [1575]
500 DATA 79,20,4D,4B,20,31,39,39,01F4         [1314]
510 DATA 31,00,01,86,DF,ED,49,21,02EE         [1817]
520 DATA 00,C0,01,00,2C,ED,B0,CD,0357         [1713]
530 DATA 03,B9,00,1A,00,00,00,00,00D6         [1256]
540 DATA END                                   [267]
```

Funktionen aufgespürt

Ein Utility für Pascal-Programmierer

Wer in Turbo-Pascal programmiert, wünscht sich sicher etwas mehr Übersicht bei seinen Programmentwicklungen. Auch im nachhinein kann es wichtig sein, über seine selbstgeschriebenen Funktionen und Prozeduren den Überblick zu behalten. Wo war doch noch einmal die Prozedur XYZ? DECL hilft Ihnen bei der Suche in Ihren Quelltexten.

DECL listet von jeder Prozedur- oder Funktionsdeklaration innerhalb eines Pascal-Quelltextes den gesamten Kopf einschließlich spezieller Schlüsselwörter wie EXTERNAL oder FORWARD auf. Es spielt dabei keine Rolle, ob im Quelltext Ihres Pascal-Programms die Schlüsselwörter groß oder klein geschrieben sind. DECL wandelt intern jedes erkannte Wort in Großbuchstaben um.

Bei der Suche nach den Schlüsselwörtern PROCEDURE und FUNCTION darf natürlich nicht innerhalb von Strings oder innerhalb von Kommentaren gesucht werden, da an diesen Stellen unter Umständen keine syntaktisch korrekten Deklarationen stehen. DECL berücksichtigt dieses Problem und ignoriert Strings und auch beide in Pascal erlaubten Kommentartypen (*...*) und {...}.

Die Ausgabe von DECL ist eine Liste aller gefundenen Prozedur- und Funktionsdeklarationen eines Pascal-Quelltextes, wobei die absolute Zeilennummer des Deklarationsbeginns in der Quelldatei angegeben wird. Zusätzlich wird abschließend die Anzahl der erkannten Prozeduren und Funktionen angegeben.

DECL unterstützt in dieser Version nur eine Ausgabe der Daten auf den Bildschirm. Eine Drucker- oder Dateiausgabe läßt sich jedoch von jedem leicht installieren.

DECL erwartet bei der Quelldatei standardmäßig die Dateiendung .PAS. Es können selbstverständlich auch andere Endungen angegeben werden. Dateien, die keine Endung besitzen, müssen für die Anwendung mit DECL mit einem Punkt schließen. Die Handhabung des Dateinamens ist äquivalent zu Turbo-Pascal.

Für erfahrene Pascal-Programmierer wird sich nach Betrachtung der ersten Abbildung die Frage erübrigen, ob DECL auch die Schachtelung von Prozeduren und Funktionen erkennt und vor allem darstellen kann.

Die Antwort ist "ja". In Pascal ist es schließlich möglich, Prozeduren innerhalb von Prozeduren, Funktionen innerhalb von Funktionen und beliebige Kombinationen dieser Strukturen ineinander zu verschachteln. DECL erkennt diese Schachtelung und stellt sie durch Einrückung in der Ausgabe dar.

DECL ist beim Aufruf ein Parameter mitzugeben: Der Name der zu durchforstenden Datei.

Wie in der Abbildung ersichtlich ist, entwickelt DECL bei der Ausgabe der Deklaration eine gewisse Eigenwilligkeit: Es schert sich nicht um das Format einer Deklaration, sondern gestaltet sie nach den folgenden Regeln neu:

Deklarationen, die nicht in einer Ausgabezeile darzustellen sind, werden über mehrere Zeilen verteilt. Im anderen Fall werden mehrzeilige Deklarationen unter Umständen in einer Zeile zusammengezogen. Diese Art der Darstellung wurde gewählt, um eine unserer Meinung nach kurze und übersichtliche Ausgabe zu erhalten.

Außerdem werden innerhalb einer Deklaration eingebettete Kommentare übergangen.

Markus Satte/rs

```
DECL v1.0     (C) 1992 Markus Satte & CPC International

Auflisten der Prozedur- und Funktionsdeklarationen in Pascal-Programmen
Datei: DECL.PAS

00039: PROCEDURE UpString (VAR source, dest: MyString);
00053: FUNCTION PosNonBlank (VAR Line: MyString;
                             Pos: INTEGER): INTEGER;
00063: PROCEDURE NumToStr (number: INTEGER; VAR txt: MyString);
00079: FUNCTION BeginOfWord (VAR Line: MyString; Pos: INTEGER;
                             BeginChar: CharSet): INTEGER;
00094: FUNCTION EndOfWord (VAR Line: MyString; Pos: INTEGER;
                           EndChar: CharSet): INTEGER;
00106: PROCEDURE NextLine;
00120: PROCEDURE StringComment;
00122: . PROCEDURE FindEnd (EndChar: CHAR);
00168: PROCEDURE GetNextWord (VAR word: MyString;
                             BegChars, EndChars: CharSet);
00195: PROCEDURE Out (txt: MyString);
00211: PROCEDURE LineNum (num: INTEGER);
00230: PROCEDURE FuncProcOut;
00238: . PROCEDURE ParamsOut;
00280: FUNCTION FileExist (name: MyString): BOOLEAN;
00294: FUNCTION ParamOk: BOOLEAN;
00313: MAIN

Gefunden wurden insgesamt:  15 Unterprogramme,
                    davon:   5 Funktionen
                      und:  10 Prozeduren.
```

Erzeugte Deklarationsreferenz, wenn DECL auf sich selbst angewendet wird.

```
(* ------------------------------------------------------ *)
(*                Der Funktionsspuerhund                   *)
(*       Amstrad CPC 6128/PCW 8256/8512/9512               *)
(*     (c) 1992 Markus Satte & CPC International           *)
(* Funktion: Auflistung aller Prozedur- und Funk-          *)
(*           tionsdeklarationen innerhalb von Pascal-*)
(*           porgrammen.                                   *)
(* ------------------------------------------------------ *)
```

```
{$P1024}            (* fuer Turbo Pascal, ermoeglicht *)
                    (* die Ausgabe-Umleitung           *)

PROGRAM Decl (Source, Output);
CONST LineWidth = 70;

TYPE  MyString    = STRING[255];
      CharSet     = SET OF CHAR;
```

```
VAR   Source                          : TEXT;
      allChars, Delimiters,
      Numbers, Letters                : CharSet;
      FileName,
      word, Line, OutLine, Blanks     : MyString;
      LinePos, LineCount, inProc,
      FuncCount, ProcCount,
      BeginEnd, RecordEnd, TabPos     : INTEGER;
      DeclEnd, ready                  : BOOLEAN;
(* --------------------------------------------- *)
(* Umwandeln eines Strings in Grossbuchstaben unter  *)
(* Verwendung der Turbo-Standard-Funktion upcase(),  *)
(* die nur fuer einzelne Zeichen definiert ist. Fuer *)
(* Nicht-Turboisten und dem ASCII-Zeichensatz:       *)
(* FUNCTION UpCase (ch: CHAR): CHAR;                 *)
(* BEGIN                                             *)
(*   UpCase:=ch;                                     *)
(*   IF ch IN ['a'..'z'] THEN                        *)
(*                UpCase:=Chr(Ord(ch)-32);           *)
(* END;                                              *)
PROCEDURE UpString (VAR source, dest : MyString);

VAR i : INTEGER;

BEGIN
  dest:=source;
  FOR i:=1 TO Length(dest) DO
                            dest[i]:=UpCase(dest[i]);
END;
(* --------------------------------------------- *)
(* ermittelt Position des ersten Nicht-Leerzeichens  *)
(* in einer Zeichenkette ab der Start-Position. Wird *)
(* keines mehr gefunden, wird eine groessere Pos. als*)
(* Laenge der Zeichenkette zurueckgegeben.           *)
FUNCTION PosNonBlank (VAR Line: MyString;
                          Pos: INTEGER): INTEGER;
BEGIN
  WHILE (Line[Pos] = ' ') AND (Pos < Length(Line)) DO
    Pos:=Succ(Pos);
  IF Line[Pos] = ' ' THEN Pos:=Succ(Pos);
  PosNonBlank:=Pos;
END;
(* --------------------------------------------- *)
(* Wandelt eine Integer-Zahl in eine Zeichenkette um:*)
PROCEDURE NumToStr (number:INTEGER; VAR txt: MyString);

VAR  ch : CHAR;

BEGIN
  txt:='';
  REPEAT
    ch:=Chr(number MOD 10 + Ord('0'));
    txt:=Concat(ch,txt);
    number:=number DIV 10;
  UNTIL number = 0;
END;
(* --------------------------------------------- *)
(* sucht in der Zeile den Anfang eines Wortes ab der *)
(* angegebenen Position. "BeginChar" ist die Menge   *)
(* der Zeichen, mit denen ein Wort anfangen darf:    *)
FUNCTION BeginOfWord (VAR Line: MyString;
                          Pos: INTEGER;
                    BeginChar: CharSet): INTEGER;
BEGIN
  WHILE (Pos <= Length(Line))
  AND (NOT(Line[Pos] IN BeginChar)) DO Pos:=Succ(Pos);
  BeginOfWord:=Pos;
END;
(* --------------------------------------------- *)
(* sucht in der Zeile das Ende eines Wortes ab der   *)
(* angegebenen Position. "EndChar" ist die Menge der *)
(* Zeichen, die ein Wort abschliessen duerfen. Bei   *)
(* Compilern, die beide UNTIL-Bedingungen testen,    *)
(* muss hier die Bereichspruefung kurz ausgeschaltet *)
(* werden.                                           *)
FUNCTION EndOfWord (VAR Line: MyString;
                        Pos: INTEGER;
                    EndChar: CharSet): INTEGER;
BEGIN
  REPEAT
    Pos:=Succ(Pos);
  UNTIL (Pos > Length(Line)) OR (Line[Pos] IN EndChar);
  EndOfWord:=Pos;
END;
(* --------------------------------------------- *)
(* eine nicht leere Zeile aus der Quelldatei lesen   *)
(* und Zeilenzaehler erhoehen:                       *)
PROCEDURE NextLine;

BEGIN
  REPEAT
    ReadLn(Source,Line); LineCount:=Succ(LineCount);
  UNTIL Eof(Source) OR (Line <> '');
END;
```

```
(* --------------------------------------------- *)
(* Ueberliesst Zeichenketten-Konstanten und Kommen-  *)
(* tare. Ist "Line[LinePos]" ein Leerzeichen, wird   *)
(* der Anfang des naechsten Textes gesucht. Beginnt  *)
(* ein Kommentar oder String, wird dieser Ueberlesen *)
(* und "Line[LinePos]" auf das den Kommentar folgende*)
(* Nicht-Leerzeichen gestellt:                       *)
PROCEDURE StringComment;

  PROCEDURE FindEnd (EndChar: CHAR);      (* klar ?! *)
  BEGIN
    REPEAT
      LinePos:=Succ(LinePos);
      IF LinePos <= Length(Line) THEN BEGIN
        IF Line[LinePos] = EndChar THEN BEGIN
          IF EndChar = '*' THEN BEGIN
            IF (LinePos < Length(Line))
            AND (Line[Succ(LinePos)] = ')') THEN BEGIN
              EndChar:=' '; LinePos:=Succ(LinePos);
            END;
          END
          ELSE
            EndChar:=' ';
        END;

      END
      ELSE IF Eof(Source) THEN  EndChar:=' '
      ELSE BEGIN
        NextLine;  LinePos:=0;
      END;
    UNTIL EndChar = ' ';
    LinePos:=PosNonBlank(Line,Succ(LinePos));
    StringComment; (*falls gleich wieder einer folgt!*)
  END;

BEGIN
  LinePos:=PosNonBlank(Line,LinePos);
  WHILE (LinePos > Length(Line)) AND NOT Eof(Source) DO
  BEGIN
    NextLine;  LinePos:=PosNonBlank(Line,1);
  END;
  IF LinePos <= Length(Line) THEN
    CASE Line[LinePos] OF
      '''' : FindEnd('''');
       'p' : FindEnd('t');
       '(' : IF LinePos < Length(Line) THEN
               IF Line[Succ(LinePos)] = '*' THEN BEGIN
                 LinePos:=Succ(LinePos);  FindEnd('*');
               END;
    END;
END;
(* --------------------------------------------- *)
(*                Das naechste Wort liefern.         *)
(* "BegChars" ist die Menge der Zeichen, mit denen   *)
(* das Wort anfangen darf, "EndChars" enthaelt die   *)
(* Menge, die nicht zu einem Wort gehoeren:          *)
PROCEDURE GetNextWord (VAR word: MyString;
                          BegChars, EndChars:CharSet);

VAR  Pos : INTEGER;

BEGIN
  word:='';
  WHILE (word = '') AND NOT Eof(Source) DO BEGIN
    StringComment;
    IF LinePos <= Length(Line) THEN
      IF Line[LinePos] IN BegChars THEN BEGIN
        Pos:=BeginOfWord(Line,LinePos,BegChars);
        LinePos:=EndOfWord(Line,Pos,EndChars);
        word:=Copy(Line,Pos,LinePos-Pos);
      END
      ELSE LinePos:=Succ(LinePos);
  END;
END;
(* --------------------------------------------- *)
(*                    Text-Ausgabe.                  *)
(* Die Ausgabe wird solange zwischengespeichert, bis *)
(* durch eine neue Ausgabe "LineWidth" ueberschritten*)
(* wird oder eine leere Zeichenkette uebergeben wird.*)
(* Nach der Ausgabe wird die naechste Ausgabe ab der *)
(* Position "TabPos" getaetigt. Bei Ueberschreitung  *)
(* von "LineWidth" wird der auszugebende Text in die *)
(* naechste Zeile uebernommen:                       *)
PROCEDURE Out (txt: MyString);

BEGIN
  IF txt = '' THEN BEGIN
    WriteLn(OutLine);  OutLine:='';  TabPos:=0;
  END
  ELSE IF Length(txt)+Length(OutLine) > LineWidth THEN
  BEGIN
    WriteLn(OutLine);  OutLine:=Copy(Blanks,1,TabPos);
  END;
```

```
  OutLine:=Concat(OutLine,txt);
END;
(* ------------------------------------------------- *)
(* Zeilennummer der Prozedur/Funktion ausgeben und   *)
(* entsprechend der Schachtelung eine Einrueckung er-*)
(* zeugen:                                           *)
PROCEDURE LineNum (num: INTEGER);

VAR  i : INTEGER;  temp: MyString;

BEGIN
  Out('');
  IF num > 0 THEN BEGIN
    NumToStr(num,temp);
    WHILE Length(temp) < 5 DO temp:=Concat('0',temp);
    Out(Concat(temp,': '));
  END
  ELSE Out('       ');
  IF inProc > 1 THEN BEGIN
    FOR i:=2 TO inProc DO Out('.');
    Out(' ');
  END;
END;
(* ------------------------------------------------- *)
(*        Prozedur/Funktions-Deklaration ausgeben:  *)
PROCEDURE FuncProcOut;

VAR temp : MyString;

  (* ---------------------------------------------- *)
  (* Parameterliste der Deklaration ausgeben. Dabei  *)
  (* werden zu lange Zeilen umgebrochen und neu for- *)
  (* matiert:                                        *)
  PROCEDURE ParamsOut;

  VAR  tempword : MyString;

  BEGIN
    Out(temp);  TabPos:=Length(OutLine)+2;  temp:=' (';
    REPEAT
      GetNextWord(word,Letters,Delimiters);
      UpString(word,tempword);
      IF tempword = 'VAR' THEN
        temp:=Concat(temp,tempword,' ')
      ELSE BEGIN
      temp:=Concat(temp,word);
      GetNextWord(word,[',',';',':','(',')'],allChars);
      temp:=Concat(temp,word);
      IF word <> ')' THEN temp:=Concat(temp,' ');
      IF word = ';' THEN BEGIN
        Out(temp);  temp:=''
      END;
      END;
    UNTIL word = ')';
    GetNextWord(word,[':',';'],allChars);
  END;

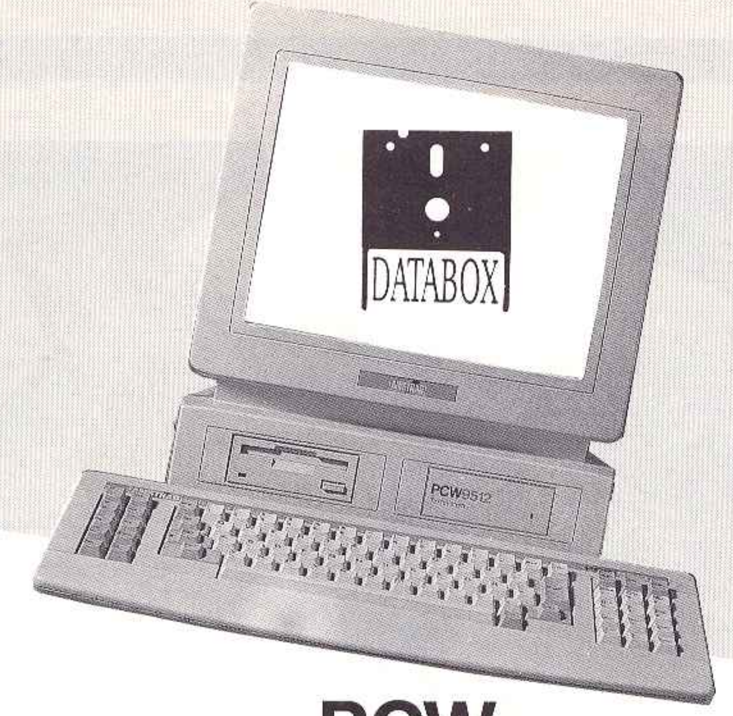
BEGIN
  LineNum(LineCount);  TabPos:=0;
  (*Deklarations-Anfang "PROC/FUNC Name ..." aufbauen*)
  temp:=Concat(word,' ');
  GetNextWord(word,Letters,Delimiters);       (* Name *)
  temp:=Concat(temp,word);
  GetNextWord(word,['(',':',';'],allChars);
  IF word = '(' THEN ParamsOut;  (* Parameterliste ? *)
  IF word = ':' THEN BEGIN          (* Funktionstyp ? *)
    GetNextWord(word,Letters,Delimiters);
    temp:=Concat(temp,': ',word);
   GetNextWord(word,[';'],allChars);
  END;
  IF word = ';' THEN temp:=Concat(temp,';');
  Out(temp);
END;
(* ------------------------------------------------- *)
(*       Test, ob angegebene Datei vorhanden ist:    *)
FUNCTION FileExist (name : MyString) : BOOLEAN;

VAR f : TEXT;

BEGIN
  Assign(f,name);
  {$I-} (*I/O-Ueberwachung durch Laufzeitsystem aus. *)
  ReSet(f);
  {$I+}                   (* ... und wieder an ! *)
  FileExist:=IOResult = 0;
  Close(f);
END;
(* ------------------------------------------------- *)
(*        pruefen, ob Parameterzeile ok ist:         *)
FUNCTION ParamOk: BOOLEAN;

BEGIN
  ParamOk:=FALSE;
  IF ParamCount = 0 THEN
WriteLn('*** Fehler: ungueltige Parameter angegeben !')
  ELSE BEGIN
    FileName:=ParamStr(1);
    IF Pos('.',FileName) = 0 THEN
      FileName:=Concat(FileName,'.PAS');
    UpString(FileName,FileName);
    IF NOT FileExist(FileName) THEN
      WriteLn('*** Fehler: Datei "',FileName,
              '" nicht gefunden !')
    ELSE
      ParamOk:=TRUE;
  END;
END;
(* ------------------------------------------------- *)
BEGIN (* Decl *)
  WriteLn;
  WriteLn('DECL v1.0     ',
          '(C) 1992 Markus Satte & CPC International');
  WriteLn;
  WriteLn('Auflisten der Prozedur- und Funktionsdekla',
          'rationen in Pascal-Programmen');
  IF ParamOk THEN BEGIN
    WriteLn('Datei: ',FileName);  WriteLn;
    allchars:=[' '..'t'];
    Numbers :=['0'..'9'];
    Letters :=['A'..'Z','a'..'z'];
    Delimiters:=allchars - Numbers - Letters - ['_'];
    Line:='';  OutLine:='';  Blanks:='';
   FOR LinePos:=1 TO 100 DO Blanks:=Concat(Blanks,' ');
    LineCount:=0;       inProc    :=0;
    BeginEnd :=0;       RecordEnd:=0;
    FuncCount:=0;       ProcCount:=0;
    LinePos  :=1;       ready:=FALSE;
    Assign(Source,FileName);
    ReSet(Source);
    WHILE NOT Eof(Source) AND NOT ready DO BEGIN
      GetNextWord(word,Letters,Delimiters);
      UpString(word,word);
      IF word = 'BEGIN' THEN BEGIN
        IF inProc = 0 THEN BEGIN
          ready:=TRUE; LineNum(LineCount); Out('MAIN');
        END

        ELSE BeginEnd:=Succ(BeginEnd);
      END
      ELSE IF word = 'CASE' THEN BEGIN
        IF RecordEnd = 0 THEN BeginEnd:=Succ(BeginEnd);
      END
      ELSE IF word = 'RECORD' THEN
        RecordEnd:=Succ(RecordEnd)
      ELSE IF (word = 'EXTERNAL') OR (word = 'FORWARD')
      THEN BEGIN
        LineNum(0);  Out(word);  inProc:=Pred(inProc);
      END
      ELSE IF word = 'END' THEN
        IF RecordEnd > 0 THEN
          RecordEnd:=Pred(RecordEnd)
        ELSE BEGIN
          BeginEnd:=Pred(BeginEnd);
          IF (inProc > 0) AND (BeginEnd = 0) THEN
            inProc:=Pred(inProc)
        END
      ELSE IF (word='PROCEDURE') OR (word='FUNCTION')
      THEN BEGIN
        inProc:=Succ(inProc);
        IF word = 'PROCEDURE' THEN
          ProcCount:=Succ(ProcCount)
        ELSE
          FuncCount:=Succ(FuncCount);
        FuncProcOut;
      END;
    END; (* WHILE *)
    Out('');
    Close(Source);
    WriteLn;
    WriteLn('Gefunden wurden insgesamt: ',
            FuncCount+ProcCount:3,' Unterprogramme,');
    WriteLn('                    davon: ',
            FuncCount:3,' Funktionen');
    WriteLn('                      und: ',
            ProcCount:3,' Prozeduren.');
    WriteLn;
  END;
  WriteLn;
END.
```



PCW DATABOX

DATABOX
ist der preiswerte Software-Service Ihrer Zeitschrift

DATABOX
enthält lauffähige Programme für Ihren Einsatz

DATABOX
lohnt sich auf jeden Fall - Monat für Monat

PCW 8256/8512/9512

Wetterstation CPC
Unsere Wetterstation aus der vergangenen CPC hat Zuwachs bekommen. Mit geringem Hard- und Softwareaufwand kann nun der PCW auch noch Temperaturdaten auswerten. Die benötigte Software finden Sie auf der DATABOX.

Patiencen
Im Heft fanden Sie "Die Schwierige". Die DATABOX kann neben dieser Kartenvariante noch mit zwei weiteren Patiencen aufwarten. Unter den Titeln "Zickzack-Kurs" sowie "Der Zopf" wird der Nervenkitzel dann um so stärker.

Trickkiste
Zum ersten Mal haben wir auch in der Trickkiste für PCW- Besitzer interessantes anzubieten. Wie wäre es zum Beispiel mit einem kleinen Trick, der es erlaubt auch unter JRT-Pascal mit Leerstrings zu arbeiten? Oder aber unser nützlicher Trick, der Ihnen zeigt, wie unter Verwendung des Betriebssystems, auch unter Turbo-Pascal das Directory eines Datenträgers eingelesen werden kann.

Einzelbezugspreis für DATABOX:
PCW – 3-Zoll-Diskette

24,– DM
Wenn Sie über den DMV-Verlag bestellen, gilt folgendes:

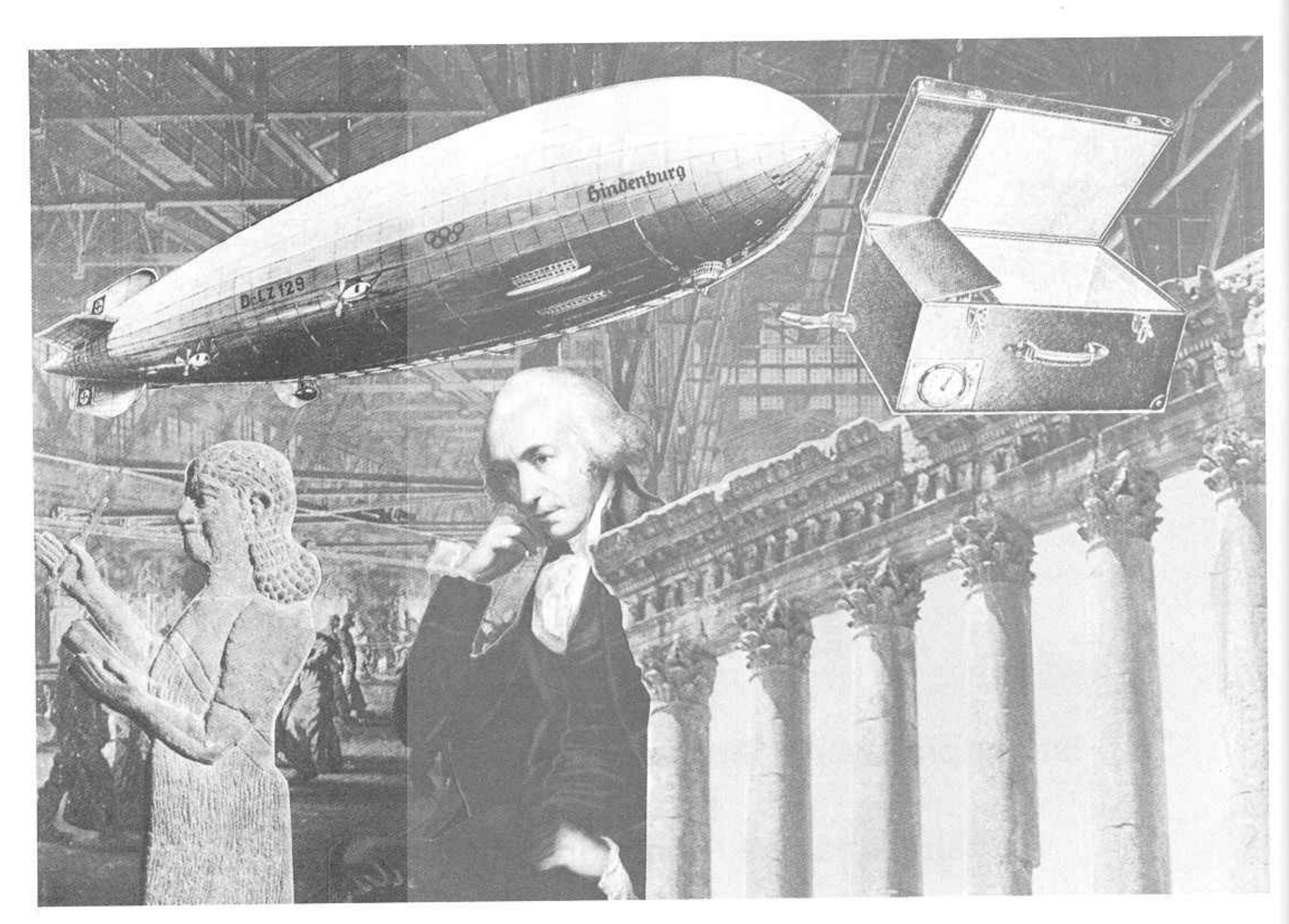
Inland:		**Ausland:**	
Einzelpreis	24,– DM	Einzelpreis	24,– DM
zzgl. Versandkosten	4,– DM	zzgl. Versandkosten	6,– DM
Endpreis	**28,– DM**	**Endpreis**	**30,– DM**

Zahlungshinweise:
Am einfachsten per Vorkasse (Verrechnungsscheck) oder als Nachnahme zuzüglich der Nachnahmegebühr. (Bei Lieferungen ins Ausland ist Nachnahme nicht möglich.)

Bitte benutzen Sie die Bestellkarte.

DMV-Verlag · Postfach 250 · 3440 Eschwege





Sag mir, wann ...

Geschichtsdaten lernen und nachschlagen mit "History"

Was – Sie wissen nicht, wann Karl der Kulleräugige die wilden Antisepten in der furchtbaren Schlacht bei Hunzburg besiegte? Könnte es am Ende sein, daß Sie ein gespaltenes Verhältnis zu geschichtlichen Daten haben? In diesem Fall (und in etlichen tausend weiteren Fällen) brauchen Sie "History", unsere CPC-Geschichtsdatenbank.

Geschichtsbücher pflegen durch viele Hände zu gehen. Der Schüler, der die eselsohrige, bekritzelte Schwarte im Schweiße seines Angesichts nach Daten von Lebens- und Regierungszeiten, Schlachten und Friedenskonferenzen befragt, findet in ihr meistens auch die Lebenszeichen vergangener Schülergenerationen, die sich mit dem gleichen Stoff abgeplagt haben. Und wie das einmal Geschehene seine Spuren unveränderlich durch die Zeit zieht, bleiben auch die alten Anmerkungen und Eselsbrücken stets die gleichen. "Drei – drei – drei, bei Issos Keilerei" – mit diesem wenig geistvollen Merkspruch etwa haben Geschichtslehrer bereits unseren Großeltern eingehämmert, was im Jahre Dreihundertdreiunddreißig Weltbewegendes passiert ist.
Auch wenn die Zeit des sturen Paukens in der Schule prinzipiell vorbei sein mag – ohne historische Daten geht es nach wie vor nicht. Allerdings ist nicht einzusehen, warum das Hantieren mit Geschichtszahlen nicht auch Spaß machen sollte. Wozu hat man schließlich seinen CPC? Und, schwuppdiwupp: Da ist das Programm "History" – die Ablösung für zerrupfte Geschichtskladden und unübersichtliche historische Kompendien. Es läuft auf allen CPCs einschließlich Plus und kommt sogar mit Kassettenspeicherung klar.

Aus deutschen Landen frisch auf den Bildschirm

Wie es sich für ein Programm aus deutschen Landen gehört, hat "History" die deutschen Umlaute samt "ß" eingebaut – nicht nur für die Bildschirmausgabe der Programmtexte, sondern auch für die Dateneingabe. So bleibt es dem Benutzer erspart, sich ständigen Zahnschmerzen bei der Beschäftigung mit "Boehmischen Feldzuegen" und ähnlichen "Merkwuerdigkeiten" auszusetzen. Sie erreichen die Umlaute im Programm über folgende Tasten:

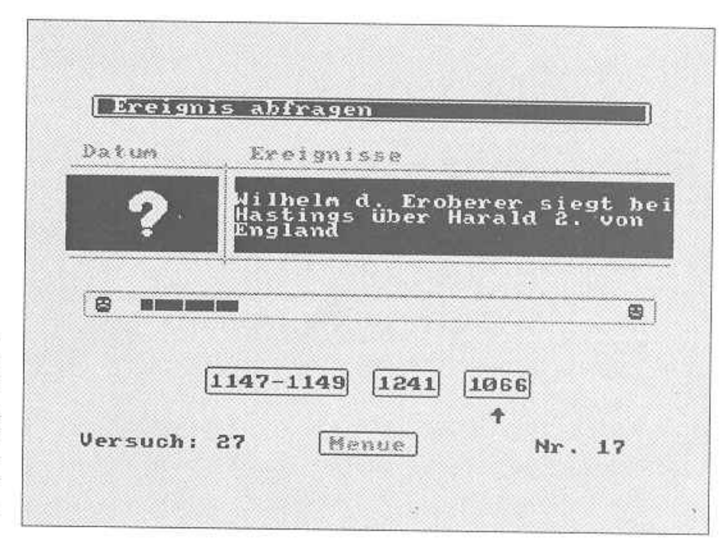
ä:	<@>
ö:	<[>
ü:	<]>
ß:	<↑>("£")

Wie Sie sehen werden, sind deutsche Umlaute auch im Programmlisting

zahlreich vertreten. Um dieses Listing abzutippen, müssen Sie daher anders vorgehen als gewohnt:
Geben Sie zunächst die Programmzeilen bis 390 ein. Speichern Sie das Teilprogramm nach erfolgter Fehlerkontrolle sicherheitshalber ab. *Starten* Sie es dann mit "`RUN`". Nach der "Ready"-Meldung stehen die deutschen Umlaute auf Ihrer Tastatur zur Verfügung. Nun machen Sie mit dem Abtippen des Listings bei Zeile 400 weiter.
Wenn Sie mit dem Checksummer arbeiten, ist die Sache etwas komplizierter:
1. Eingeben: "`PRINT HIMEM`". Ausgegebenen Wert notieren. Dann Checksummer laden und starten.
2. Listing bis Zeile 390 abtippen und abspeichern. Dann eingeben: "`MEMORY` XXX". Für "XXX" schreiben Sie den vorher notierten Wert.
3. Starten des Programms mit "`RUN`". Anschließend den Rest ab Zeile 400 zunächst *ohne Hilfe des Checksummers* abtippen. Nach getaner Tipparbeit "History.Bas" abspeichern.
4. Reset durchführen (<Shift>+<Control>+<Esc>). Dann noch einmal Checksummer laden und starten. Anschließend das abgespeicherte Listing "History.Bas" laden, aber *nicht* starten.
5. Geben Sie ein: "`|CHECK,0,256`". Stoppen Sie die Ausgabe nach Zeile 440 mit <Esc>. Überprüfen Sie die Programmzeilen samt darunterstehenden Prüfsummen auf dem Bildschirm. Anschließend lassen Sie die Ausgabe durch Druck einer Taste weiterlaufen und stoppen sie nach dem nächsten Block erneut. Korrigiert wird dann wie gewohnt (zweimal <Esc>, "`EDIT` Zeilennr." oder Übernahme vom Bildschirm mit dem <Copy>-Cursor, Programmzeile abschließen mit <Return>). Stimmen alle angezeigten Prüfsummen mit den im Heft abgedruckten überein, speichern Sie "History.Bas" erneut ab.

Bequem durch die Jahrhunderte

Nach dem Start meldet sich das Programm mit dem Hauptmenü, das neun Optionen zur Auswahl stellt. Ebenso wie in allen Untermenüs wählen Sie hier die gewünschte Option mit den Cursortasten "rauf"/"runter" und bestätigen Ihre Wahl mit <Copy>.
Eingabe: Daten und Ergnisse werden erfaßt. Es gibt dazu zwei Eingabefelder; jede Eingabe wird mit <Return> abgeschlossen. Eine Datumseingabe läßt sich rückgängig machen, solange noch kein Ereignis angegeben wurde: Ein "L" im Ereignisfenster führt zurück zur Datumseingabe.
Damit das Programm die Daten später sortieren kann, müssen die Zeitangaben in einer festgelegten Form erfolgen. Je nachdem, ob ein bestimmter Tag, ein Monat, ein Jahr oder ein Zeitraum erfaßt werden soll, sind folgende Eingaben richtig:
"15000 v", "1992", "07.1992", "23.07.1992" und "1992-1993".
Jedes Zeitdatum vor Christi Geburt wird mit einem nachgestellten "v" versehen – mit oder ohne Leerzeichen davor.
Ändern: Es erscheint ein Untermenü, in dem Sie sich für das Löschen, Einfügen, Ändern oder Suchen einzelner Einträge entscheiden.
Anschauen: Beim Durchblättern lernen Sie die eingegebenen und/oder geladenen Geschichtsdaten einer Datei.



Wie bei der Führerscheinprüfung: Beim Abfragen braucht kein Text eingegeben zu werden. Der Benutzer wählt die richtige Lösung mit den Cursortasten aus.

Auswahl durch "Ankreuzen"

Ereignis abfragen: Das Programm gibt ein Ereignis vor und stellt drei mögliche Zeitangaben zur Auswahl. Wählen Sie das zutreffende Datum mit den Cursortasten "rechts"/"links" und bestätigen Sie mit <Copy>. Der grafische Balken in der Mitte des Bildschirms gibt über den momentanen Erfolgsstand Auskunft.
Datum abfragen: das entsprechende Gegenstück mit vorgegebenem Datum und wählbarem Ereignis.
Daten speichern/laden: Es erscheint ein Untermenü mit verschiedenen Dateioperationen. Beim "Mergen" wird der Inhalt einer Geschichtsdatei zu den bereits im Speicher des Computers befindlichen Daten hinzugeladen. Ein so zusammengefügter Datenbestand läßt sich dann als Gesamtdatei wieder abspeichern. Dateinamen werden ohne Erweiterung eingegeben – das Programm setzt automatisch die Erweiterung ".HIS" an den Namen der gewünschten Geschichtsdatei.
Die Option "Speed" kommt nur zum Zuge, wenn man Geschichtsdaten auf Kassette speichern will: Sie erlaubt es, die Speichergeschwindigkeit auf 1000 (1) oder 2000 (2) Bits pro Sekunde zu setzen.
Lexikon: Hier sucht das Programm zu einer Zeitangabe des Benutzers das passende Ereignis heraus – sofern es im Speicher eines findet. Die Benutzereingabe und das gespeicherte Datum des Ereignisses müssen genau übereinstimmen, auch was die Schreibweise betrifft.
Sortieren: Setzt neue Einträge innerhalb des Datenbestands im Speicher an die richtige Stelle. Der Sortiervorgang läßt sich mit <C> starten. <Z> läßt den gespeicherten Datenbestand komplett durchlaufen, <l> verringert die Durchlaufgeschwindigkeit, <h> hält die Anzeige an (weiter mit beliebiger Taste) und <e> bricht sie ab.
Ende: Ein Untermenü erlaubt es, entweder die im Speicher vorhandenen Daten zu löschen oder ins Basic zurückzukehren.
Übrigens: Bezieher der DATABOX-Diskette zu diesem Heft haben gegenüber den fleißigen Abtippern den kleinen Vorteil, daß sie zu "History" bereits eine Beispieldatei "Daten.His" mit rund 100 Einträgen aus der Weltgeschichte mitgeliefert bekommen.

(Armin Müller/sz)

```
100 ' HISTORY                                    [1027]
110 ' (c) 1992 Reiner Mueller                    [1332]
120 '          & CPC International               [2118]
130 IF HIMEM<40000 THEN 330                      [632]
140 '*** Fragezeichen ***                        [1399]
150 SYMBOL AFTER 175                             [1433]
160 SYMBOL 175,0,&3,&F,&1F,&3F,&7F,&7F,&7F [2621]
170 SYMBOL 176,&FC,&FF,&FF,&FF,&FF,&FF,&8F [2267]
,&7
180 SYMBOL 177,0,0,&C0,&E0,&F0,&F8,&F8,&F8 [2265]
190 SYMBOL 178,&7F,&7F,&3E,0,0,0,0,1       [1928]
200 SYMBOL 179,&7,&7,&F,&1F,&3F,&7F,&FF,&F [2019]
F
210 SYMBOL 180,&F8,&F8,&F8,&F0,&E0,&C0,&80 [2925]
220 SYMBOL 181,1,1                         [977]
230 SYMBOL 182,&FE,&FC,&F8,0,&70,&F8,&F8,& [2700]
70
240 '*** Umlaute ***                       [876]
250 SYMBOL 183,&DB,&3C,&66,&66,&7E,&66,&66 [2183]
:' Ae
260 SYMBOL 184,&BB,&6C,&C6,&C6,&C6,&6C,&38 [3002]
:' Oe
270 SYMBOL 185,&C3,&66,&66,&66,&66,&66,&3C [2169]
:' Ue
280 SYMBOL 186,&CC,0,&78,&C,&7C,&CC,&76: ' [2230]
 ae
290 SYMBOL 187,&66,0,&3C,&66,&66,&66,&3C:' [2981]
 oe
300 SYMBOL 188,&66,0,&66,&66,&66,&66,&3E:' [1884]
 ue
310 SYMBOL 189,&3C,&66,&66,&6C,&66,&66,&6C [2522]
,&60
320 OPENOUT"d":MEMORY HIMEM-1:CLOSEOUT     [2613]
330 KEY DEF 26,1,186,183                   [810]
340 KEY DEF 17,1,187,184                   [1016]
350 KEY DEF 19,1,188,185                   [1194]
360 KEY DEF 24,1,189,163,0                 [1376]
370 SPEED KEY 20,1                         [1372]
380 ENT 2,100,2,20                         [763]
390 PRINT CHR$(23)                         [1145]
400 INK 0,2:INK 1,24:INK 2,14:INK 3,8      [587]
410 BORDER 11                              [950]
420 DEFINT a-i,k-w,y,z                     [1242]
430 j=20:p=13:n=1:wa=100:dm=105:'*** dm =  [3745]
max. Anzahl von Daten
440 DIM dtm$(dm),day(dm),monat(dm),jahr(dm [6604]
),ind(dm),erg$(dm),nd$(dm),ne$(dm),kontrol
le(j),wahl1$(9),wahl2$(6),wahl3$(6)
450 text3$="Zuwenig Daten vorhanden !"     [1517]
460 FOR a=0 TO 9:READ wahl1$(a):NEXT       [1612]
470 DATA "Menü",Eingabe,"Ändern",Anschauen [9297]
,Ereignis abfragen,Datum abfragen,Daten sp
eichern/laden,Lexikon,Sortieren,Ende
480 FOR a=1 TO 6:READ wahl2$(a):NEXT       [3288]
490 DATA "Menü","Löschen",Suchen,"Ändern", [5736]
Suchen,"Einfügen"
500 FOR a=1 TO 6:READ wahl3$(a):NEXT       [2774]
510 DATA "Menü",Load,Save,Merge,Cat,Speed  [3007]
520 ON BREAK GOSUB 5580                    [595]
530 '### Menue ###                         [779]
540 MODE 1                                 [506]
550 PAPER#0,0                              [1312]
560 PEN 1                                  [549]
570 POKE &B1C8,0:POKE &B1CF,207:POKE &B1D0 [1939]
,118
580 LOCATE 6,7:PRINT"History"              [2540]
590 POKE &B1C8,1:POKE &B1CF,136:POKE &B1D0 [2427]
,68
600 MOVE 80,280:DRAW 555,280               [1249]
610 PEN 2                                  [548]
620 MOVE 1,1:DRAW 1,398,1:DRAW 638,398:DRA [2539]
W 638,1:DRAW 1,1
630 pp=p:p=12:GOSUB 5180:p=pp              [1141]
640 FOR a=1 TO 9:LOCATE 11,a+12:PRINT wahl [2379]
1$(a):NEXT
650 MOVE 120,216:DRAW 500,216,3:DRAW 500,5 [4063]
5:DRAW 120,55:DRAW 120,216
660 LOCATE 9,p:PRINT CHR$(243)" "CHR$(24)w [2743]
ahl1$(p-12)CHR$(24)
670 LOCATE 7,24:PRINT"Es sind"an"Daten vor [3958]
handen!"
680 a$=INKEY$:IF a$="" THEN 680            [1392]
690 a=ASC(a$)                              [1036]
700 LOCATE 9,p:PRINT"  "wahl1$(p-12)       [1040]
710 IF a=224 THEN ON p-12 GOTO 770,1140,19 [3533]
30,2170,2880,3570,4200,4720,5040
720 IF a=240 THEN p=p-1:IF p<13 THEN p=21  [2295]
730 IF a=241 THEN p=p+1:IF p>21 THEN p=13  [1702]
740 LOCATE 9,p:PRINT CHR$(243)" "CHR$(24)w [2743]
ahl1$(p-12)CHR$(24)
750 GOTO 680                               [460]
760 '### Eingabe ###                       [509]
770 GOSUB 5150                             [853]
780 MOVE 166,317:DRAW 166,308              [2081]
790 MOVE 2,86:DRAW 636,86                  [1397]
800 MOVE 166,90:DRAW 166,22                [1287]
810 MOVE 2,25:DRAW 636,25                  [922]
820 WINDOW#1,1,10,7,19                     [1245]
830 WINDOW#2,12,40,7,19                    [1531]
840 WINDOW#3,1,10,21,23                    [1469]
850 WINDOW#4,12,40,21,23                   [1584]
860 PAPER#1,2:PAPER#2,2                    [2027]
870 PAPER#3,3:PAPER#4,3                    [1962]
880 PEN#1,1:PEN#2,1                        [901]
890 PEN#3,1:PEN#4,1                        [1528]
900 CLS#1:CLS#2:CLS#3:CLS#4                [1769]
910 LOCATE 15,25:PRINT"L-Löschen   E-Ende" [1724]
920 an=an+1:IF an>dm-1 THEN BORDER 4:LOCAT [5905]
E#7,11,1:PEN#7,3:PRINT#7,"  Speicher voll!
";:PEN#7,1:CALL &BB18:BORDER 11:GOTO 1100
930 GOSUB 1120                             [859]
940 CLS#3:CLS#4                            [695]
950 LINE INPUT#3,"",eingabe1$              [3065]
960 IF eingabe1$="" THEN 940               [467]
970 IF LOWER$(eingabe1$)="e" THEN 1100     [1829]
980 'dtm$(an)=LEFT$(eingabe1$,10)          [1615]
990 dtm$(an)=eingabe1$:'datzeil=INT(LEN(ei [2633]
ngabe1$)/10)
1000 GOSUB 1120                            [859]
1010 CLS#4                                 [386]
1020 LINE INPUT#4,"",eingabe2$             [2240]
1030 IF eingabe2$="" THEN 1010             [757]
1040 IF LOWER$(eingabe2$)="e" THEN 1100    [1202]
1050 IF LOWER$(eingabe2$)="l" THEN 940     [1605]
1060 erg$(an)=LEFT$(eingabe2$,86)          [1782]
1070 CLS#3:CLS#4                           [695]
1080 i=an:GOSUB 5600:' Scroll              [1796]
1090 GOTO 920                              [332]
1100 dtm$(an)="":an=an-1                   [795]
1110 GOTO 540                              [425]
1120 LOCATE 3,25:PRINT"Nr.";an:RETURN      [1527]
1130 '### Aendern ###                      [881]
1140 MODE 1                                [506]
1150 pp=3:h=6                              [205]
1160 GOSUB 5180                            [879]
1170 WINDOW#1,1,10,15,23                   [1236]
1180 WINDOW#2,12,40,15,23                  [1586]
1190 PAPER#1,2:PAPER#2,2                   [2027]
1200 PEN#1,1:PEN#2,1                       [901]
1210 CLS#1:CLS#2                           [576]
1220 LOCATE 1,13:PEN 3:PRINT" Datum     Er [3789]
eignisse":PEN 1
1230 MOVE 2,187:DRAW 636,187               [1386]
1240 MOVE 168,192:DRAW 168,20              [1207]
1250 MOVE 2,25:DRAW 636,25                 [922]
1260 LOCATE 6,7:PRINT"Nr.";n AND an>0      [3015]
1270 FOR a=1 TO 6:LOCATE 28,a+5:PRINT wahl [2583]
2$(a):NEXT
1280 LOCATE 35,8:PRINT CHR$(244)           [2050]
1290 LOCATE 35,9:PRINT CHR$(207)           [2055]
1300 LOCATE 35,10:PRINT CHR$(245)          [1857]
1310 LOCATE 28,6:PRINT CHR$(24)wahl2$(1)CH [3217]
R$(24)SPC(5)CHR$(242):' Pfeil
1320 MOVE 425,324:DRAW 598,324:DRAW 598,22 [3515]
0:DRAW 425,220:DRAW 425,324
1330 PRINT#1,dtm$(n):PRINT#2,erg$(n)       [2052]
1340 '+++ Abfrage +++                      [695]
1350 a$=INKEY$:IF a$="" THEN 1350          [1474]
1360 a=ASC(a$)                             [1036]
1370 LOCATE 37,h:PRINT" "                  [829]
1380 IF a=224 AND h=6 THEN 540             [1721]
1390 IF a=224 AND an>0 THEN ON h-6 GOSUB 1 [4017]
470,1610,1670,1750,1810
1400 LOCATE 28,h:PRINT wahl2$(h-5)         [592]
1410 IF a=240 THEN h=h-1:IF h<6 THEN h=11  [2968]
1420 IF a=241 THEN h=h+1:IF h>11 THEN h=6  [1986]
1430 LOCATE 6,7:PRINT"Nr."n AND an>0 SPC(4 [2322]
)
1440 LOCATE 28,h:PRINT CHR$(24)wahl2$(h-5) [2399]
CHR$(24):LOCATE 37,h:PRINT CHR$(242)
1450 GOTO 1350                             [460]
1460 '+++ Loeschen +++                     [1017]
1470 nn=n                                  [263]
1480 LOCATE#2,3,7:PRINT#2,"Löschen...   [J [2232]
/N]
1490 a$=INKEY$:IF a$="" THEN 1490          [1439]
1500 IF LOWER$(a$)<>"j" THEN LOCATE#2,3,7: [3278]
PRINT#2,SPC(20):RETURN
1510 IF an=1 THEN dtm$(n)="":erg$(n)="":an [3460]
=0:n=1:GOTO 1630
1520 an=an-1:fl=1                          [672]
```

```
1530 FOR i=n TO an                          [660]
1540 dtm$(i)=dtm$(i+1):erg$(i)=erg$(i+1)    [2269]
1550 NEXT                                   [350]
1560 dtm$(an+1)="":erg$(an+1)=""            [1455]
1570 IF an<n THEN nn=0:n=1                  [813]
1580 GOSUB 1630                             [861]
1590 fl=0:nn=nn+1:LOCATE 6,7:PRINT"Nr."nn   [2600]
AND an>0 SPC(4):GOTO 1480
1600 '+++ aufwaerts zeigen +++              [701]
1610 n=n+1                                  [140]
1620 IF n>an THEN n=1                       [79]
1630 CLS#1:CLS#2                            [576]
1640 PRINT#1,dtm$(n):PRINT#2,erg$(n)        [2052]
1650 RETURN                                 [555]
1660 '+++ Daten aendern +++                 [1574]
1670 LOCATE#1,1,7:LINE INPUT#1,"",eingabe1  [2065]
$
1680 LOCATE#2,1,7:LINE INPUT#2,"",eingabe2  [2955]
$
1690 IF NOT eingabe1$="" THEN dtm$(n)=LEFT  [2777]
$(eingabe1$,10)
1700 IF NOT eingabe2$="" THEN erg$(n)=LEFT  [2853]
$(eingabe2$,86)
1710 CLS#1:CLS#2                            [576]
1720 PRINT#1,dtm$(n):PRINT#2,erg$(n)        [2052]
1730 RETURN                                 [555]
1740 '+++ abwaerts zeigen +++               [936]
1750 n=n-1                                  [104]
1760 IF n<1 THEN n=an                       [935]
1770 CLS#1:CLS#2                            [576]
1780 PRINT#1,dtm$(n):PRINT#2,erg$(n)        [2052]
1790 RETURN                                 [555]
1800 '+++ Einfuegen +++                     [875]
1810 IF an+1>dm-1 THEN RETURN               [900]
1820 LOCATE#1,1,7:LINE INPUT#1,"",eingabe1  [2065]
$
1830 IF eingabe1$="" THEN RETURN            [385]
1840 LOCATE#2,1,7:LINE INPUT#2,"",eingabe2  [2955]
$
1850 IF eingabe2$="" THEN RETURN            [1044]
1860 n=n+1                                  [140]
1870 FOR i=an TO n STEP -1                  [504]
1880 dtm$(i+1)=dtm$(i):erg$(i+1)=erg$(i)    [1372]
1890 NEXT                                   [350]
1900 GOSUB 1690:an=an+1:LOCATE 6,7:PRINT"N  [3050]
r.";n
1910 GOTO 1810                              [377]
1920 '### Anschauen ###                     [342]
1930 GOSUB 5150                             [853]
1940 MOVE 166,317:DRAW 166,308              [2081]
1950 MOVE 2,42:DRAW 636,42,3                [1030]
1960 LOCATE 1,24:PRINT CHR$(24)"COPY-Menü/  [6438]
A-Anfang/Z-zurück/Taste-weiter"CHR$(24);
1970 WINDOW#1,1,10,7,22                     [1258]
1980 WINDOW#2,12,40,7,22                    [1524]
1990 PAPER#1,0:PAPER#2,0                    [2107]
2000 PEN#1,1:PEN#2,1                        [901]
2010 IF st>=an THEN st=0                    [237]
2020 b=0                                    [312]
2030 st=st+1                                [342]
2040 i=st:GOSUB 5600:' Scroll               [850]
2050 b=b+1                                  [569]
2060 IF st=an THEN PRINT CHR$(7);:GOTO 209  [1975]
0
2070 IF b=4 THEN 2090                       [719]
2080 GOTO 2030                              [365]
2090 a$=INKEY$:IF a$="" THEN 2090           [1510]
2100 CLS#1:CLS#2                            [576]
2110 IF a$=CHR$(224) THEN st=st-b:GOTO 540  [1462]
2120 IF LOWER$(a$)="a" THEN st=0            [1783]
2130 IF LOWER$(a$)="z" THEN st=st-b:IF st-  [2967]
4>=0 THEN st=st-4
2140 IF st=an THEN st=0                     [712]
2150 GOTO 2020                              [359]
2160 '### Ereignisse abfragen ###           [846]
2170 ver=0:schritt=0:zufall=10:xpo=80:ypo=  [1614]
162
2180 GOSUB 5150                             [853]
2190 WINDOW#1,1,10,7,11                     [1253]
2200 WINDOW#2,12,40,7,11                    [1539]
2210 WINDOW#3,3,38,19,21                    [900]
2220 WINDOW#4,8,33,22,22                    [1405]
2230 PAPER#1,2:PAPER#2,2                    [2027]
2240 PAPER#3,0:PAPER#4,0                    [2067]
2250 PEN#1,0:PEN#2,0:PEN#3,2                [2208]
2260 CLS#1:CLS#2                            [576]
2270 LOCATE 3,15:PRINT CHR$(225)TAB(38)CHR  [3138]
$(224)
2280 MOVE 20,180:DRAW 620,180,3:DRAW 620,1  [3524]
55:DRAW 20,155:DRAW 20,180
2290 MOVE 166,317:DRAW 166,212              [2081]
2300 MOVE 4,217:DRAW 634,217                [1609]
2310 MOVE 280,30:TAG:PRINT"Menü";:TAGOFF    [1964]
2320 MOVE 268,37:DRAW 356,37,1:DRAW 356,13  [2253]
:DRAW 268,13:DRAW 268,37
2330 LOCATE 2,24:PRINT"Versuch:":LOCATE 32  [3473]
,24:PRINT"Nr."
2340 IF an<3 THEN PRINT#3,TAB(6)text3$:GOT  [3072]
O 2760
2350 ERASE kontrolle:DIM kontrolle(j)       [1531]
2360 FOR anzahl=1 TO an                     [371]
2370 GOSUB 5230:' Daten auswaehlen          [1386]
2380 LOCATE 32,24:PRINT"Nr.";anzahl         [2429]
2390 CLS#1:CLS#2                            [576]
2400 PRINT#2:PRINT#2,erg$(zahl1)            [564]
2410 auswahl$(1)=dtm$(zahl1)                [870]
2420 auswahl$(2)=dtm$(zahl2)                [1391]
2430 auswahl$(3)=dtm$(zahl3)                [1520]
2440 LOCATE#1,5,2:PRINT#1,CHR$(175)CHR$(17  [3710]
6)CHR$(177)
2450 LOCATE#1,5,3:PRINT#1,CHR$(178)CHR$(17  [1935]
9)CHR$(180)
2460 LOCATE#1,5,4:PRINT#1,CHR$(181)CHR$(18  [3012]
2)
2470 le=LEN(auswahl$(1)+auswahl$(2)+auswah  [2287]
l$(3))+4
2480 LOCATE 1,17:PRINT SPACE$(240);:abstd=  [3037]
320-le*16/2:TAG:MOVE abstd,95
2490 FOR b=1 TO 3                           [1127]
2500 PRINT auswahl$(wechsel(b));:GOSUB 284  [2503]
0:xp(b)=le
2510 NEXT b                                 [378]
2520 altzahl=wechsel(1):h=63:v=2            [741]
2530 PEN 1                                  [549]
2540 GOSUB 2680:PRINT CHR$(240);:' Pfeil s  [3085]
etzen
2550 '+++ Steuerung +++                     [910]
2560 a$=INKEY$:IF a$="" THEN 2560           [1504]
2570 a=ASC(a$)                              [1036]
2580 IF a=224 THEN 2710                     [1007]
2590 IF a=241 THEN 2800                     [907]
2600 CLS#4                                  [386]
2610 IF a=242 THEN v=v-1:IF v<1 THEN v=3    [1990]
2620 IF a=243 THEN v=v+1:IF v>3 THEN v=1    [2101]
2630 ON v GOSUB 2670,2680,2690              [1746]
2640 PRINT CHR$(240);                       [1053]
2650 GOTO 2560                              [315]
2660 '+++ Pfeile setzen +++                 [2064]
2670 MOVE abstd+xp(1)/2-8,h:RETURN          [2352]
2680 MOVE abstd+xp(1)+24+xp(2)/2,h:RETURN   [2081]
2690 MOVE abstd+xp(1)+56+xp(2)+xp(3)/2,h:R  [1984]
ETURN
2700 '+++ Eingabe pruefen +++               [1533]
2710 ver=ver+1:TAGOFF:LOCATE 2,24:PRINT"Ve  [11368]
rsuch:";ver:TAG:IF auswahl$(wechsel(v))=dt
m$(zahl1) THEN CLS#1:PRINT#1:PRINT#1,dtm$(
zahl1) ELSE SOUND 2,2200,50,15:GOTO 2560
2720 GOSUB 5520:' Balken zeichnen           [1666]
2730 GOSUB 5440:' Melodie                   [1626]
2740 warte=1:AFTER wa,1 GOSUB 5580:WHILE w  [3726]
arte=1:WEND
2750 TAGOFF:NEXT anzahl                     [2498]
2760 CLS#4:TAG::MOVE 312,63:PRINT CHR$(241  [1617]
);
2770 CALL &BB18                             [389]
2780 TAGOFF:GOTO 540                        [1223]
2790 '+++ zurueck zum Menue +++             [1615]
2800 CLS#4:MOVE 312,63:PRINT CHR$(241);     [2417]
2810 a$=INKEY$:IF a$="" THEN 2810           [1431]
2820 IF a$=CHR$(224) THEN 2780 ELSE 2570    [1760]
2830 '+++ Rechteck zeichnen +++             [2233]
2840 le=LEN(auswahl$(wechsel(b)))*16        [1662]
2850 MOVER 0,8:DRAWR -le-4,0:DRAWR 0,-28:D  [3800]
RAWR le+4,0:DRAWR 0,28:MOVER 32,-8
2860 RETURN                                 [555]
2870 '### Jahreszahlen abfragen ###         [1508]
2880 ver=0:schritt=0:zufall=10:xpo=80:ypo=  [1789]
195
2890 GOSUB 5150                             [853]
2900 WINDOW#1,1,10,7,9                      [1123]
2910 WINDOW#2,12,40,7,9                     [1392]
2920 WINDOW#3,11,39,16,24                   [1139]
2930 PAPER#1,2:PAPER#2,2:PAPER#3,0          [2336]
2940 PEN#1,0:PEN#2,0:PEN#3,1                [2201]
2950 CLS#1:CLS#2:CLS#3                      [1230]
2960 MOVE 166,317:DRAW 166,245,3            [2257]
2970 MOVE 4,248:DRAW 634,248                [1582]
2980 PEN 3:FOR a=18 TO 21:LOCATE 5,a:PRINT  [4613]
 MID$("Menü",a-17,1):NEXT a:PEN 1
2990 MOVE 55,52:DRAW 55,136,1:DRAW 88,136:  [2951]
DRAW 88,52:DRAW 55,52
```

```
3000 MOVE 152,165:DRAW 630,165,3:DRAW 630, [3617]
8:DRAW 152,8:DRAW 152,165
3010 LOCATE 3,13:PRINT CHR$(225)TAB(38)CHR [2501]
$(224)
3020 MOVE 20,215:DRAW 620,215:DRAW 620,185 [3104]
:DRAW 20,185:DRAW 20,215
3030 LOCATE 1,16:PRINT"Vs:":LOCATE 1,24:PR [1732]
INT"Nr."
3040 IF an<3 THEN LOCATE#3,2,5:PRINT#3,tex [3152]
t3$:GOTO 3500
3050 MOVE 152,112:DRAW 630,112 [1083]
3060 MOVE 152,64:DRAW 630,64 [1434]
3070 ERASE kontrolle:DIM kontrolle(j) [1531]
3080 FOR anzahl=1 TO an [371]
3090 GOSUB 5230:' Daten auswaehlen [1386]
3100 LOCATE 1,24:PRINT"Nr.";anzahl [1172]
3110 CLS#1:CLS#2 [576]
3120 PRINT#1,dtm$(zahl1); [1273]
3130 auswahl$(1)=erg$(zahl1) [864]
3140 auswahl$(2)=erg$(zahl2) [1194]
3150 auswahl$(3)=erg$(zahl3) [850]
3160 LOCATE#2,12,1:PRINT#2,CHR$(175)CHR$(1 [2653]
76)CHR$(177)
3170 LOCATE#2,12,2:PRINT#2,CHR$(178)CHR$(1 [2543]
79)CHR$(180)
3180 LOCATE#2,12,3:PRINT#2,CHR$(181)CHR$(1 [2853]
82)
3190 '+++ Daten zeigen +++ [919]
3200 CLS#3:a=0 [463]
3210 FOR b=1 TO 3 [1127]
3220 LOCATE#3,1,b+a:PRINT#3,auswahl$(wechs [2685]
el(b))
3230 a=a+2 [724]
3240 NEXT b [378]
3250 MOVE 160,112:DRAW 622,112,3 [1725]
3260 MOVE 160,64:DRAW 622,64 [803]
3270 altzahl=wechsel(1):PEN 1 [490]
3280 LOCATE 8,20:PRINT CHR$(243):h=20 [1101]
3290 '+++ Steuerung +++ [910]
3300 a$=INKEY$:IF a$="" THEN 3300 [1437]
3310 a=ASC(a$) [1036]
3320 IF a=224 THEN 3400 [829]
3330 LOCATE 8,h:PRINT" " [715]
3340 IF a=242 THEN 3530 [1490]
3350 IF a=240 THEN h=h-3:IF h<17 THEN h=23 [3095]
3360 IF a=241 THEN h=h+3:IF h>23 THEN h=17 [1030]
3370 LOCATE 8,h:PRINT CHR$(243) [1566]
3380 GOTO 3300 [453]
3390 '+++ Eingabe pruefen +++ [1533]
3400 pruef=h-16 [749]
3410 ver=ver+1:LOCATE 1,16:PRINT"Vs:";ver [2872]
3420 IF pruef=4 THEN pruef=2 [986]
3430 IF pruef=7 THEN pruef=3 [1058]
3440 IF auswahl$(wechsel(pruef))=erg$(zahl [8160]
1) THEN CLS#2:PRINT#2,erg$(zahl1) ELSE SOU
ND 2,2200,50,15:GOTO 3300
3450 GOSUB 5520:' Balken zeichnen [1666]
3460 GOSUB 5440:' Melodie [1626]
3470 warte=1:AFTER wa,1 GOSUB 5580:WHILE w [3726]
arte=1:WEND
3480 LOCATE 8,h:PRINT" " [715]
3490 NEXT anzahl [969]
3500 LOCATE 8,20:PRINT CHR$(242) [1194]
3510 CALL &BB18:GOTO 540 [1236]
3520 '+++ zurueck zum Menue +++ [1615]
3530 LOCATE 8,20:PRINT CHR$(242) [1194]
3540 a$=INKEY$:IF a$="" THEN 3540 [1453]
3550 IF a$=CHR$(224) THEN 540 ELSE LOCATE [2762]
8,20:PRINT" ":GOTO 3310
3560 '### Load,Save,Merge,Cat ### [1396]
3570 MODE 1:GOSUB 5180 [895]
3580 h=1 [419]
3590 WINDOW#0,2,39,13,24 [1020]
3600 WINDOW#1,5,12,5,10 [1480]
3610 PAPER#0,3:PAPER#1,2 [2651]
3620 CLS#0:CLS#1 [761]
3630 PEN#0,1:PEN#1,0 [1166]
3640 MOVE 60,236:DRAW 60,338,1:DRAW 194,33 [3246]
8:DRAW 194,236:DRAW 60,236
3650 MOVE 8,214:DRAW 630,214:DRAW 630,8:DR [3348]
AW 8,8:DRAW 8,214
3660 MOVE 1,1:DRAW 1,398:DRAW 638,398:DRAW [2768]
 638,1:DRAW 1,1
3670 FOR a=1 TO 6:LOCATE#1,3,a:PRINT#1,wah [3447]
l3$(a):NEXT
3680 LOCATE#1,1,h:PRINT#1,CHR$(243)" "CHR$ [3793]
(24)wahl3$(h)CHR$(24)
3690 '+++ Abfrage +++ [695]
3700 a$=INKEY$:IF a$="" THEN 3700 [1484]
3710 a=ASC(a$) [1036]
3720 LOCATE#1,1,h:PRINT#1," " [1708]
3730 IF a=224 THEN CLS#0:ON h GOTO 540,380 [2923]
0,3880,3940,4020,4120
3740 LOCATE#1,3,h:PRINT#1,wahl3$(h) [1746]
3750 IF a=240 THEN h=h-1:IF h<1 THEN h=6 [2622]
3760 IF a=241 THEN h=h+1:IF h>6 THEN h=1 [2181]
3770 LOCATE#1,1,h:PRINT#1,CHR$(243)" "CHR$ [3793]
(24)wahl3$(h)CHR$(24)
3780 GOTO 3700 [501]
3790 '+++ Daten Laden +++ [1247]
3800 GOSUB 4050 [1015]
3810 ERASE dtm$,erg$ [517]
3820 DIM dtm$(dm),erg$(dm) [990]
3830 OPENIN name$ [1050]
3840 INPUT#9,an [841]
3850 FOR i=1 TO an:LINE INPUT#9,dtm$(i):LI [2857]
NE INPUT#9,erg$(i):NEXT
3860 CLOSEIN:GOTO 3680 [1043]
3870 '+++ Daten speichern +++ [736]
3880 GOSUB 4050 [1015]
3890 OPENOUT name$ [700]
3900 PRINT#9,an [1000]
3910 FOR i=1 TO an:PRINT#9,dtm$(i):PRINT#9 [3611]
,erg$(i):NEXT
3920 CLOSEOUT:GOTO 3680 [1535]
3930 '+++ Merge +++ [596]
3940 GOSUB 4050:alt=an [1003]
3950 OPENIN name$ [1050]
3960 INPUT#9,an [841]
3970 IF alt+an>dm-1 THEN an=alt:PRINT:PRIN [6864]
T:PRINT"Speicher zu klein !":GOTO 4000
3980 FOR i=1 TO an:LINE INPUT#9,dtm$(i+alt [2723]
):LINE INPUT#9,erg$(i+alt):NEXT
3990 an=an+alt [625]
4000 CLOSEIN:GOTO 3680 [1043]
4010 '+++ Cat +++ [599]
4020 CAT [96]
4030 GOTO 3680 [664]
4040 '+++ Eingabe des Dateinamens +++ [1920]
4050 PRINT:PRINT"Bitte Dateiname eingeben: [3214]
"
4060 PRINT:INPUT name$:PRINT [1080]
4070 IF name$="" THEN GOTO 3680 [1136]
4080 IF LEN(name$) >10 THEN 4050 [1594]
4090 name$=LOWER$(name$)+".His" [766]
4100 RETURN [555]
4110 '+++ Geschwindigkeit aendern +++ [2601]
4120 PRINT:PRINT"Bitte Geschwindigkeit wäh [5714]
len (1/2)"
4130 PRINT:INPUT a [417]
4140 IF a=0 THEN 3680 [445]
4150 sd=a-1 [401]
4160 IF sd<0 OR sd>1 THEN 4120 [390]
4170 SPEED WRITE sd [1188]
4180 GOTO 3680 [664]
4190 '### Lexikon ### [814]
4200 MODE 1 [506]
4210 GOSUB 5180 [879]
4220 WINDOW#1,1,10,14,18 [1281]
4230 WINDOW#2,12,40,14,18 [1421]
4240 PAPER#1,3:PAPER#2,3 [2094]
4250 PEN#1,1:PEN#2,1:PEN 2 [1710]
4260 MOVE 2,200:DRAW 636,200 [1646]
4270 MOVE 2,102:DRAW 636,102 [1048]
4280 MOVE 167,98:DRAW 167,204 [1048]
4290 LOCATE 2,12:PRINT"Datum"SPC(6)"Ereign [4664]
is"
4300 CLS#1:CLS#2 [576]
4310 MOVE 150,70:DRAW 490,70:DRAW 490,40:D [3021]
RAW 150,40:DRAW 150,70
4320 LOCATE 2,8:PRINT"Datum: ";:PEN 1: PRI [2752]
NT CHR$(143)
4330 LOCATE 15,22:PEN 1:PRINT "COPY = Menü [3508]
":x=8:eingabe$=""
4340 '+++ Eingabe +++ [721]
4350 a$=INKEY$:IF a$="" THEN 4350 [1498]
4360 a=ASC(a$) [1036]
4370 IF a=224 THEN 540 [1006]
4380 IF a=13 THEN 4520 [1107]
4390 IF a=127 THEN 4460 [1309]
4400 IF LEN(eingabe$)>20 THEN 4350 [1633]
4410 x=x+1 [853]
4420 LOCATE x,8 [701]
4430 PRINT CHR$(a)CHR$(143) [1371]
4440 eingabe$=eingabe$+CHR$(a) [1074]
4450 GOTO 4350 [547]
4460 IF LEN(eingabe$)=0 THEN 4350 [1970]
4470 LOCATE x,8:PRINT CHR$(143)" " [1720]
4480 x=x-1 [849]
4490 eingabe$=LEFT$(eingabe$,x-8) [1978]
```

```
4500 GOTO 4350                                    [547]
4510 '+++ Ereignis suchen +++                     [1351]
4520 LOCATE 9,8:PRINT CHR$(143)SPACE$(22); [8022]
:el=LEN(eingabe$):hilf$="":GOSUB 4650:LOCA
TE 14,22:PRINT "Moment bitte!"
4530 FOR i=1 TO el                                [583]
4540 hilf1$=MID$(eingabe$,i,1):IF hilf1$<> [4122]
" " THEN hilf$=hilf$+hilf1$
4550 NEXT                                         [350]
4560 eingabe$=hilf$                               [1508]
4570 FOR such=1 TO an                             [1125]
4580 el=LEN(dtm$(such)):hilf$=""                  [1518]
4590 FOR i=1 TO el                                [583]
4600 hilf1$=MID$(dtm$(such),i,1):IF hilf1$ [3801]
<>" " THEN hilf$=hilf$+hilf1$
4610 NEXT                                         [350]
4620 IF eingabe$=hilf$ THEN 4670                  [1300]
4630 NEXT:LOCATE 12,22:PRINT"Nichts  gefun [5072]
den !":warte=1:AFTER wa,1 GOSUB 5580
4640 WHILE warte=1:WEND:GOSUB 4650:GOTO 43 [3310]
30
4650 LOCATE 12,22:PRINT SPACE$(18)                [1605]
4660 RETURN                                       [555]
4670 GOSUB 4650:CLS#1:CLS#2                       [2604]
4680 PRINT#1:PRINT#2                              [788]
4690 PRINT#1,dtm$(such);:PRINT#2,erg$(such [1853]
)
4700 GOTO 4330                                    [551]
4710 '### Sortieren ###                           [807]
4720 GOSUB 5150                                   [853]
4730 WINDOW#1,1,10,7,18                           [1246]
4740 WINDOW#2,12,40,7,18                          [1528]
4750 PAPER#1,2:PEN#1,1                            [2040]
4760 PAPER#2,2 :PEN#2,1                           [1707]
4770 CLS#1:CLS#2                                  [576]
4780 MOVE 4,105:DRAW 635,105,3:MOVE 168,30 [3015]
8:DRAW 168,318
4790 LOCATE 1,24:PEN 1:PRINT CHR$(24)"COPY [3967]
-Menü/C-sort/Z-zeig/L-lgm/H-halt/E-."CHR$(
24);
4800 MOVE 4,98:DRAW 635,98,1:DRAW 635,37:D [2896]
RAW 4,37:DRAW 4,98
4810 a$=INKEY$:IF a$="" THEN 4810                 [1542]
4820 IF a$=CHR$(224) THEN 540                     [907]
4830 IF LOWER$(a$)="z" THEN 4920                  [981]
4840 IF LOWER$(a$)="c" THEN 4860                  [2132]
4850 GOTO 4810                                    [448]
4860 CLS#1:CLS#2:PRINT#2," Bitte warten !" [2689]
4870 EVERY 50,1 GOSUB 5020                        [909]
4880 GOSUB 5670                                   [917]
4890 re=REMAIN(1)                                 [906]
4900 PRINT#2:PRINT#2:PRINT#2," fertig !!"   [2479]
4910 GOTO 4810                                    [448]
4920 CLS#1:CLS#2                                  [576]
4930 FOR i=1 TO an                                [549]
4940 GOSUB 5600                                   [954]
4950 a$=INKEY$                                    [278]
4960 IF LOWER$(a$)="l" THEN IF ls=700 THEN [1867]
 ls=1 ELSE ls=700
4970 IF LOWER$(a$)="e" THEN 4810                  [1666]
4980 IF LOWER$(a$)="h" THEN CALL &BB18            [1097]
4990 FOR w=0 TO ls:NEXT                           [1358]
5000 NEXT                                         [350]
5010 GOTO 4810                                    [448]
5020 PRINT#2,".";:RETURN                          [863]
5030 '### Ende ###                                [633]
5040 MODE 1                                       [506]
5050 GOSUB 5180:PEN 2                             [1386]
5060 MOVE 1,1:DRAW 1,398:DRAW 638,398:DRAW [2768]
 638,1:DRAW 1,1
5070 LOCATE 5,10:PRINT"COPY - Menü"               [1724]
5080 LOCATE 8,12:PRINT"L - Daten löschen"         [2202]
5090 LOCATE 8,14:PRINT"E - Programm verlas [2848]
sen"
5100 a$=INKEY$:IF a$="" THEN 5100                 [1492]
5110 IF LOWER$(a$)="e" THEN MODE 1:PEN 1:P [2541]
RINT:END
5120 IF  LOWER$(a$)="l" THEN LOCATE 5,20:I [4383]
NPUT"Daten wirklich löschen  J/N";f$:IF LO
WER$(f$)="j" THEN  RUN
5130 GOTO 540                                     [425]
5140 '*** Menuewahl zeigen ***                    [1731]
5150 MODE 1                                       [506]
5160 LOCATE 1,5:PEN 3:PRINT" Datum"SPC(6)" [2973]
Ereignisse":PEN 1
5170 MOVE 4,312:DRAW 634,312                      [864]
5180 WINDOW#7,3,38,2,2:PAPER#7,2:CLS#7            [2044]
5190 MOVE 26,365:DRAW 26,386,1:DRAW 612,38 [2501]
6:DRAW 612,365:DRAW 26,365
5200 PEN#7,0:PRINT#7," ";wahl1$(p-12);:PEN [3130]
#7,1
5210 RETURN                                       [555]
5220 '*** Daten auswaehlen ***                    [2196]
5230 IF an<10 THEN zufall=an                      [1328]
5240 IF an>10 AND an-anzahl=an MOD 10 THEN [2408]
 zufall=zufall+an MOD 10
5250 zahl1=INT(RND*zufall)+schritt+1              [1405]
5260 FOR i=1 TO anzahl-schritt                    [2225]
5270 IF kontrolle(i)=zahl1 THEN 5250              [1745]
5280 NEXT i                                       [375]
5290 kontrolle(anzahl-schritt)=zahl1              [1110]
5300 IF anzahl MOD 10=0 AND zufall=10 THEN [3624]
 schritt=schritt+10
5310 zahl2=INT(RND*zufall)+schritt+1              [2454]
5320 IF zahl2=zahl1 THEN 5310                     [846]
5330 zahl3=INT(RND*zufall)+schritt+1              [2035]
5340 IF zahl3=zahl2 OR zahl3=zahl1 THEN 53 [3563]
30
5350 '*** Wechsel der Daten ***                   [1097]
5360 wechsel(1)=INT(RND*3)+1                      [1239]
5370 IF wechsel(1)=altzahl THEN 5360              [1577]
5380 wechsel(2)=INT(RND*3)+1                      [1961]
5390 IF wechsel(2)=wechsel(1) THEN 5380           [1822]
5400 wechsel(3)=INT(RND*3)+1                      [1462]
5410 IF wechsel(3)=wechsel(2) OR wechsel(3 [3188]
)=wechsel(1) THEN 5400
5420 RETURN                                       [555]
5430 '*** Melodie ***                             [1728]
5440 RESTORE 5500                                 [715]
5450 FOR ml=1 TO 12                               [716]
5460 READ ton                                     [738]
5470 SOUND 2,ton,10,14                            [1875]
5480 NEXT                                         [350]
5490 RETURN                                       [555]
5500 DATA 358,301,358,426,301,358,426,301, [2495]
338,301,253,225
5510 '*** Balken zeichnen ***                     [2651]
5520 FOR g=1 TO 480\an STEP 2                     [1480]
5530 MOVE xpo+g,ypo:DRAWR 0,10,1                  [1281]
5540 NEXT                                         [350]
5550 xpo=xpo+480/an                               [1200]
5560 RETURN                                       [555]
5570 '*** Warten beenden ***                      [2000]
5580 warte=0:RETURN                               [1165]
5590 '*** Scroll ***                              [968]
5600 PRINT#1,dtm$(i)                              [585]
5610 abstd=LEN(erg$(i))\29+1                      [1331]
5620 FOR scroll=1 TO abstd:PRINT#1:NEXT           [2921]
5630 PRINT#2,erg$(i):PRINT#2                      [1267]
5640 datzeil=INT(LEN(dtm$(i))/10)                 [1269]
5650 FOR scroll=1 TO datzeil:PRINT#2:NEXT         [2765]
5660 RETURN                                       [555]
5670 'Sortieren                                   [223]
5680 FOR i=1 TO an:d$=dtm$(i)                     [1210]
5690 day(i)=0:monat(i)=0                          [1547]
5700 dlang=LEN(d$)                                [1062]
5710 ppos=INSTR(d$,"."):IF ppos<>0 THEN pp [3844]
os2=INSTR(ppos+1,d$,"."):ELSE ppos2=0
5720 IF ppos=0 AND ppos2=0 THEN jahr(i)=VA [5389]
L(d$):IF INSTR(d$,"v")<>0 THEN jahr(i)=jah
r(i)* -1:GOTO 5760
5730 IF ppos2<>0 THEN jahr(i)=VAL(RIGHT$(d [9621]
$,dlang-ppos2)):day(i)=VAL(LEFT$(d$,ppos))
:IF INSTR(d$,"v") THEN jahr(i)=jahr(i)* -1
5740 IF ppos2=0 THEN jahr(i)=VAL(RIGHT$(d$ [5195]
,dlang-ppos)):IF INSTR(d$,"v") THEN jahr(i
)=jahr(i)* -1
5750 IF ppos2=0 THEN monat(i)=VAL(LEFT$(d$ [7305]
,ppos)):ELSE monat(i)=VAL(MID$(d$,ppos+1,p
pos2-ppos-1))
5760 NEXT                                         [350]
5770 FOR i=1 TO an                                [549]
5780 ind=1                                        [283]
5790 FOR such=1 TO an                             [1125]
5800 IF jahr(such)>jahr(ind) THEN ind=such [3407]
:GOTO 5850
5810 IF jahr(such)<jahr(ind) THEN 5850            [1090]
5820 IF monat(such)>monat(ind) THEN ind=su [4184]
ch:GOTO 5850
5830 IF monat(such)<monat(ind) THEN 5850          [2628]
5840 IF day(such)>day(ind) THEN ind=such          [1405]
5850 NEXT                                         [350]
5860 ind(i)=ind:jahr(ind)=-1E+38:monat(ind [2510]
)=0:day(ind)=0
5870 NEXT                                         [350]
5880 FOR i=1 TO an                                [549]
5890 nd$(an+1-i)=dtm$(ind(i)):ne$(an+1-i)= [3785]
erg$(ind(i))
5900 NEXT                                         [350]
5910 FOR i=1 TO an                                [549]
5920 dtm$(i)=nd$(i):erg$(i)=ne$(i)                [1306]
5930 NEXT                                         [350]
5940 RETURN                                       [555]
```

Think

Bombenjagd mit Grips

Bevor eine versteckte Bombe entschärft werden kann, muß man sie erst einmal finden. In unserem Fall gehört eine Menge Scharfsinn und Logik dazu, damit es bei der Jagd nach den Bomben nicht zu bösen Überraschungen kommt. Machen Sie mit bei "Think", einem alle einfachen wie eindrucksvollen Denkspiel für alle CPCs.

Ziel des Spiels "Think" ist, in einem Spielfeld, das aus 36, 64, 100 oder 144 Feldern besteht, eine festgelegte Anzahl von Bomben zu finden und die entsprechenden Felder zu markieren. Die Anzahl der Felder und die Anzahl der Bomben hängen dabei vom gewählten Spiellevel ab. Dabei gilt:

Level	Felder	Bomben
Beginner	36	4
Amateur	64	9
Expert	100	20
Professional	144	48

Um die Bomben zu finden, kann man mit einem Cursor jedes Feld auf dem Spielfeld anspringen. Ein so gewähltes Feld kann auf zwei verschiedene Weisen behandelt werden. Ist man der Meinung, auf dem Feld befindet sich eine Bombe, sperrt man es mit einem Totenkopf, indem man die Leertaste drückt. Andernfalls deckt man das Feld durch <COPY> auf. Findet sich unter einem mit <COPY> aufgedeckten Feld eine Bombe, ist das Spiel unweigerlich zu Ende. In diesem Fall werden alle Felder offengelegt, und man kann die Positionen der Bomben erkennen.

Ist das aufgedeckte Feld "sauber", erscheint in seiner linken oberen Ecke eine Ziffer, die die Anzahl der direkt an das Feld angrenzenden Bomben zeigt. Diese Zahl bezieht sich auf alle Bomben rund um das gewählte Feld. Ist keine Zahl zu sehen, so sind alle Felder um das gewählte Feld herum bombenfrei, auch die diagonalen Nachbarn.

Ein bereits aufgedecktes Feld kann logischerweise nicht mehr markiert werden. Die Sperrung mittels Totenkopf läßt sich jedoch durch nochmaliges Drükken der Leertaste rückgängig machen.

Gesteuert wird per Menü

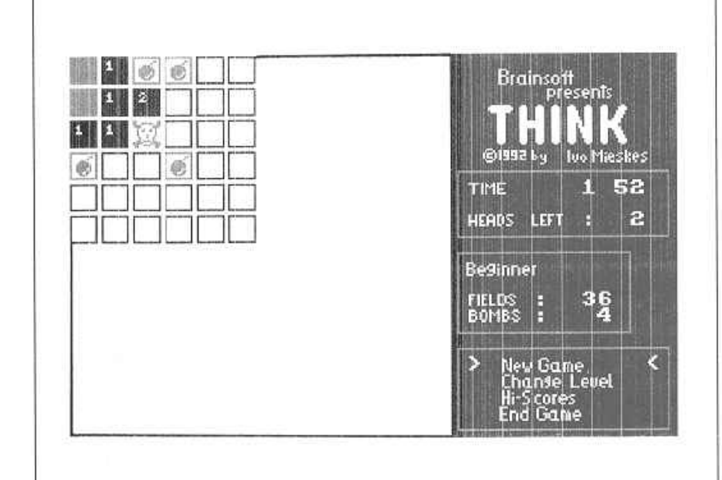
Während des Spiels läuft eine Uhr mit, die im Kontrollkasten auf der rechten Bildschirmseite unter dem Eintrag "time" eingeblendet ist. Hat man alle Bomben gefunden und als Freigabesignal <RETURN> gedrückt, stoppt die Zeit. Die schnellsten Bombensucher können sich in der High-Score-Tabelle verewigen.

Über ein kleines Menü wird das gesamte Spielgeschehen kontrolliert. Die vier Menüpunkte haben folgende Bedeutung:

New Game – startet ein neues Spiel
Change Level – wechselt den Schwierigkeitsgrad
Hi-Scores – eröffnet einen Blick auf die besten Plazierungen
End Game – bringt den Bombensucher auf den Boden des BASIC zurück

Die einzelnen Menüpunkte werden mit den Cursortasten angewählt und mit <COPY> aktiviert. Die folgende Tabelle zeigt alle Tasten und deren Funktion innerhalb des Spiels "Think".

Taste	Bedeutung
Cursortasten	Steuerung des Spielcursors und der Pfeile im Hauptmenü
<COPY>	dient zum Aufdecken eines Feldes und zum Anklicken einer Option im Hauptmenü
<SPACE>	kennzeichnet ein Feld mit einem Totenkopf oder löscht die Totenkopfmarkierung
<RETURN>	beendet das Spiel, wenn man meint, alle Bomben markiert zu haben. Es wird die Kontrollfunktion aktiviert, die die markierten Bomben mit den tatsächlich gesetzten vergleicht. Die Zeit wird gestoppt, und man kann sich gegebenenfalls in die High-Score-Liste eintragen
<DEL>	dient zum Abbrechen des Spiels. Es werden alle Bomben aufgedeckt, und man gelangt zurück ins Hauptmenü



"Think" ist eine Herausforderung an das Logikverständnis. Gefährliche Bomben müssen geortet werden. Verschiedene Levels vom Beginner ...

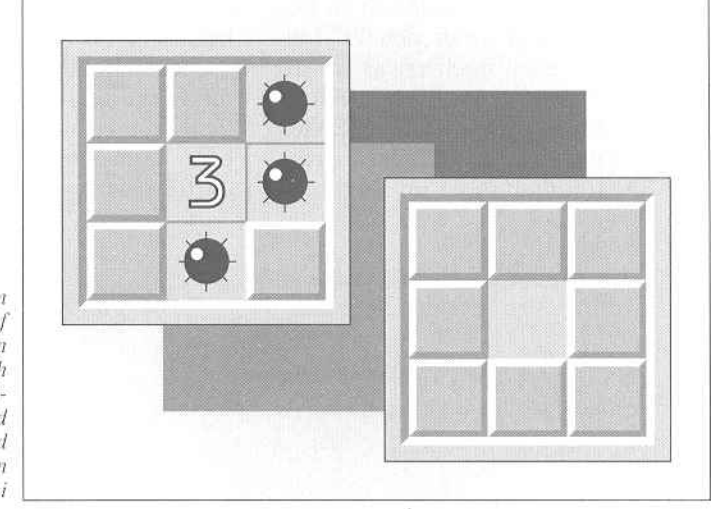
... bis zum Professional bringen eine spannende Langzeitunterhaltung mit sich

Zwei Listings sorgen für Bombenunterhaltung

"Think" besteht aus zwei Listings. Das eine heißt THINK.PRG, das andere SCORE.BAS. Letzteres ist ein kleines BASIC-Programm, das vor dem ersten Spiel gestartet werden muß. SCORE.BAS richtet eine ASCII-Datei namens THINK.SCO auf der Diskette ein, die die besten Ergebnisse für die Ewigkeit festhält. Diese Datei wird vom Hauptprogramm gebraucht. THINK.PRG ist dank des leicht verständlichen BASIC-Codes in vielen Bereichen zu modifizieren. So lassen sich beispielsweise die Farben in der Zeile 800 den individuellen Wünschen anpassen. Wer Zeit beim Abtippen sparen will, kann die REM-Zeilen weglassen, sie werden von GOTO- oder GOSUB-Befehlen nicht direkt angesprungen.

Im Falle einer Fehlermeldung

Wer ohne Diskettenlaufwerk auskommen muß, wird zu Programmbeginn vom Rechner eine Fehlermeldung erhalten."Think" erwartet die High-Score-Tabelle von der Diskette. In diesem Fall reicht es, den GOSUB-Befehl in Zeile 790 zu streichen. Das Spiel läuft dann zwar ohne High-Score-Tabelle, aber es läuft.
Wenn Sie vor dem ersten Start von "Think" das Zusatzprogramm SCORE.BAS nicht ausführen lassen, beschwert sich der Rechner, daß er die



Eine Ziffer in einem Feld bezieht sich auf alle angrenzenden Felder einschließlich der diagonalen Nachbarn. Besitzt ein Feld keine Ziffer, so sind alle angrenzenden Felder bombenfrei

High-Score-Tabelle nicht finden kann. Abhilfe: Starten von SCORE.BAS.
Im "Professional-Level" muß der gute alte CPC sehr viele Berechnungen durchführen. Dadurch wird unter anderem die Zeit des Spielfeldaufbaus verlängert. Es tut sich einige Zeit nichts auf dem Bildschirm. Etwas Geduld ist angesagt, nicht das Drücken der <ESC>-Taste. Auch die Berechnungen der Bombenanzahl benötigen in den hohen Levels etwas Zeit, auch hier gilt: Geduld ist besser als Abbruch.

Langzeit-Logik-Unterhaltung

In der Regel läßt sich "Think" ohne großes Risiko lösen. Verschiedene Kniffe und Strategien entblößen immer mehr vom Spielfeld. Nur in seltenen Fällen, wie beim Spielstart, muß man blind ins Blaue zielen – mit dem Risiko, eine Bombe zu treffen. Um den Geist von "Think" besser einfangen zu können, geben wir einige kleine Tips: Felder ohne Ziffern sind die beliebtesten Felder. Um sie herum können alle angrenzenden Felder gefahrlos geöffnet werden. Liegen um ein Feld mit einer 1 bereits sieben aufgedeckte Felder, muß im achten eine Bombe versteckt sein. Die Umgebungen aller "1"-Felder, die an ein Bombenfeld angrenzen, können gefahrlos enttarnt werden.

Kleines "Who's Who?" der Variablen

Der Übersicht halber haben wir zum Abschluß noch einmal alle Variablen und ihre Bedeutung aufgelistet. Das erleichtert ein Verändern des Programms:

a,b,d,l,m,q,s,z	Zählervariablen
low, high, hi	legen Grenzwerte in einem DIMensionierten Feld fest
cl, mi, sec, ti	Zeitvariablen
t$, t	Tastaturabfragen
name$	High-Score-Variablen
l$	Level-Variable
bm	Anzahl der zu suchenden Bomben im Spiel
xm, ym	doppelte Breite und Höhe des Spielfelds
x,y	aktuelle Cursorposition
kx, ky	Totenkopfkoordinaten
bx, by	Bombenkoordinaten
d(x,y)	Feldvariable (sie ist 0, wenn das Feld noch nicht aufgedeckt wurde und 1 im anderen Fall)

Übrigens: Es soll ja vorkommen, daß der eine oder andere unserer Leser auch

Warum nicht mal BASIC?

Ivo Mieskes über sein Programm "Think"

Der Grund, warum ich dieses Spiel geschrieben habe, ist nicht nur die Spielidee, die mir gefallen hat, sondern vielmehr die Tatsache, daß immer weniger Spiele in reinem BASIC-Code verfaßt werden.
Oft hört man, BASIC sei viel zu langsam. Doch diesen "Schönheitsfehler" kann man nicht selten durch geschickte Programmierung umgehen. Sieht man sich das Programm "Think" an, so fallen zum Beispiel die auf den ersten Blick umständlich programmierten Symboldefinitionen oder die vielen PRINT- und LOCATE-Anweisungen auf. Natürlich könnte man alle Daten schön sauber in DATA-Zeilen schreiben, dies würde aber die Ablaufgeschwindigkeit vervielfachen. Hier sieht man, daß die konventionellen Programmiermethoden die "sauberen" in puncto Geschwindigkeit bei weitem übertreffen können.
Dieses Programm ist absichtlich auf Schnelligkeit getrimmt und zeigt damit, daß auch BASIC "pur" eine durchaus taugliche Programmiersprache sein kann.

in der Welt der IBM-kompatiblen PCs heimisch ist. Wer unter den PC-Vertrauten dann auch noch etwas für die Mammut-Benutzeroberfläche Microsoft Windows übrig hat, kennt vielleicht "Minesweeper". Dieses grafische Spielprogramm ist im Lieferumfang von Windows 3.1 enthalten und erfreut sich bei vielen "Windowianern" großer Beliebtheit – denn auch mit teuren Büro-Rechnerklötzen mag man ja nicht immer nur ernsthafte Dinge treiben. Dem "Minesweeper"-Kundigen dürfte eine ungemein starke Ähnlichkeit mit unserem CPC-"Think" auffallen – will sagen: Wer sich zur fröhlichen Bombensuche aufmacht, braucht dazu nicht unbedingt immer die teuerste Computer-Umgebung!...

(Ivo Mieskes/sz)

```
100 ' THINK.BAS ' Hauptprogramm                [1196]
110 ' CPC 664/6128(plus)                       [1532]
120 ' (c) 1992 Ivo Mieskes                     [1279]
130 '         & CPC International              [2118]
140 '                                          [117]
150 ' --- Zeichensatz und Symbole definier [2399]
en
160 '                                          [117]
170 SYMBOL AFTER 135                           [1457]
180 SYMBOL 135,0,0,0,140,82,94,80,76:SYMBO [4257]
L 136,0,0,0,115,74,74,74,74
190 SYMBOL 137,0,0,29,36,165,157,5,57:SYMB [3748]
OL 138,0,224,144,147,228,151,148,227
200 SYMBOL 139,0,240,128,135,228,132,132,2 [4827]
44:SYMBOL 140,0,4,4,28,164,164,164,156
210 SYMBOL 141,0,28,34,32,38,34,34,28:SYMB [3383]
OL 142,0,0,0,115,146,146,146,122
220 SYMBOL 143,0,0,0,70,169,175,168,166:SY [4750]
MBOL 144,0,96,80,86,100,68,68,68
230 SYMBOL 145,0,1,2,98,150,146,146,98:SYM [4346]
BOL 146,0,0,0,99,148,242,129,102
240 SYMBOL 147,0,4,0,52,69,37,21,100:SYMBO [4035]
L 148,0,0,0,206,41,41,41,201
250 SYMBOL 149,0,1,1,57,73,73,73,61:SYMBOL [3963]
 150,0,112,64,69,101,66,69,117
260 SYMBOL 151,0,0,0,99,84,103,68,67:SYMBO [3866]
L 152,0,2,2,55,162,162,34,34
270 SYMBOL 153,0,48,72,75,122,74,74,74:SYM [3616]
BOL 154,0,0,0,71,169,169,169,167
280 SYMBOL 155,0,32,32,102,41,47,40,166:SY [3996]
MBOL 156,0,0,0,75,74,74,74,50
290 SYMBOL 157,0,72,74,72,122,74,74,74:SYM [2789]
BOL 158,0,14,16,16,204,2,2,28
300 SYMBOL 159,0,0,0,51,68,68,68,51:SYMBOL [3173]
 160,0,0,0,51,164,167,164,35
310 SYMBOL 161,0,0,0,24,160,144,8,48:SYMBO [3921]
L 162,0,58,66,67,66,66,66,58
320 SYMBOL 163,0,0,0,142,82,82,82,79:SYMBO [4136]
L 165,0,0,1,114,74,73,72,75
330 SYMBOL 166,0,0,128,152,164,188,160,24: [4454]
SYMBOL 167,0,16,16,16,16,16,16,30
340 SYMBOL 168,0,0,0,101,149,245,133,98:SY [3990]
MBOL 169,0,2,2,50,74,122,66,51
350 SYMBOL 170,0,68,100,84,85,85,77,68:SYM [3387]
BOL 171,0,0,0,145,81,213,10,138
360 SYMBOL 172,0,14,17,16,19,17,17,14:SYMB [3337]
OL 173,0,0,0,57,73,73,73,61
370 SYMBOL 174,0,0,0,163,84,87,84,83:SYMBO [3729]
L 175,0,0,0,0,128,128,0,0
380 SYMBOL 176,0,117,69,101,69,69,69,0:SYM [3436]
BOL 177,0,209,17,145,17,17,221,0
390 SYMBOL 178,0,198,41,36,34,41,198,0:SYM [4193]
BOL 179,0,113,74,114,74,74,113,0
400 SYMBOL 180,0,145,91,85,81,81,145,0:SYM [4738]
BOL 181,0,113,74,113,72,74,113,0
410 SYMBOL 182,0,128,64,0,128,64,128,0:SYM [3700]
BOL 183,0,75,74,74,123,74,75,0
420 SYMBOL 184,0,153,37,37,61,37,165,0:SYM [4685]
BOL 185,0,142,80,72,68,66,156,0
430 SYMBOL 188,0,142,136,136,140,136,238,0 [3804]
:SYMBOL 189,0,238,132,132,196,132,132,0
440 SYMBOL 190,0,125,17,17,17,17,17,0:SYMB [3589]
OL 191,0,69,109,85,69,69,69,0
450 SYMBOL 192,0,224,0,0,192,0,224,0:SYMBO [3743]
L 193,0,128,128,202,170,198,2,12
460 SYMBOL 194,0,32,32,42,42,42,36,0:SYMBO [3273]
L 195,0,2,3,66,162,162,66,0
470 SYMBOL 196,0,32,105,162,43,42,41,0:SYM [4138]
BOL 197,0,2,26,162,147,10,178
480 SYMBOL 198,0,0,145,170,57,160,147:SYMB [5040]
OL 199,0,0,128,0,0,128,0
490 SYMBOL 200,0,0,15,31,31,15,0,0:SYMBOL  [3668]
201,0,0,240,248,248,240,0,0
500 SYMBOL 202,0,0,255,255,255,255,60,60:S [4420]
YMBOL 203,60,60,60,60,60,60,60,60
510 SYMBOL 204,60,60,60,60,60,24,0,0:SYMBO [4880]
L 205,60,60,63,63,63,63,60,60
```

```
520 SYMBOL 206,60,60,252,252,252,252,60,60 [3999]
:SYMBOL 207,0,0,24,60,60,60,60,60
530 SYMBOL 208,0,0,24,60,60,62,62,62:SYMBO [3266]
L 209,63,63,63,61,61,61,60,60
540 SYMBOL 210,60,60,188,188,188,252,252,2 [4134]
52:SYMBOL 211,124,124,124,60,60,24,0,0
550 SYMBOL 212,0,0,12,30,30,60,60,120:SYMB [3912]
OL 213,120,240,240,224,224,240,240,120
560 SYMBOL 214,120,60,60,30,30,12,0,0:SYMB [4869]
OL 215,0,0,15,31,31,15,1,1
570 SYMBOL 216,0,0,252,254,254,252,224,224 [3298]
:SYMBOL 217,1,1,1,1,1,1,1,1
580 SYMBOL 218,224,224,224,224,224,224,224 [4788]
,224:SYMBOL 219,1,1,1,1,1,0,0,0
590 SYMBOL 220,224,224,224,224,224,192,0,0 [4922]
:SYMBOL 221,0,0,230,148,148,228,132,128
600 SYMBOL 222,0,0,99,148,242,129,102,0:SY [4469]
MBOL 223,0,0,51,74,122,66,50,0
610 SYMBOL 224,16,56,147,84,82,81,86,0:SYM [4451]
BOL 225,240,136,136,243,138,138,138,242
620 SYMBOL 226,0,1,0,57,73,73,73,61:SYMBOL [5073]
 227,0,0,0,113,74,73,72,75
630 SYMBOL 228,0,0,0,153,36,36,164,24:SYMB [4772]
OL 229,0,80,144,184,144,144,144,144
640 SYMBOL 230,0,0,0,192,128,128,128,128:S [2722]
YMBOL 231,255,255,255,251,243,251,251,241
650 SYMBOL 232,255,255,255,249,246,253,251 [3938]
,240:SYMBOL 233,255,255,255,241,254,249,25
4,241
660 SYMBOL 234,255,255,255,246,246,240,254 [4813]
,254:SYMBOL 235,255,255,255,240,247,241,25
4,241
670 SYMBOL 236,255,255,255,249,247,241,246 [4762]
,249:SYMBOL 237,255,255,255,240,254,253,25
1,251
680 SYMBOL 238,255,255,255,249,246,249,246 [4837]
,249:SYMBOL 239,0,127,64,64,64,64,64,64
690 SYMBOL 240,0,254,2,2,2,2,2,2:SYMBOL 24 [3662]
1,64,64,64,64,64,64,127,0
700 SYMBOL 242,2,2,2,2,2,2,254,0:SYMBOL 24 [4156]
3,0,127,64,64,64,67,71,79
710 SYMBOL 244,0,254,2,26,34,194,226,242:S [3464]
YMBOL 245,79,79,79,70,67,64,127,0
720 SYMBOL 246,242,210,178,98,194,2,254,0: [2567]
SYMBOL 247,0,96,231,248,56,18,22,16
730 SYMBOL 248,0,6,231,31,28,40,104,8:SYMB [4012]
OL 249,16,8,8,27,124,115,48,0
740 SYMBOL 250,136,144,16,216,62,206,12,0: [4640]
SYMBOL 251,255,255,255,255,255,255,255,255

750 SYMBOL 252,0,187,170,187,136,187,0,0:S [5117]
YMBOL 253,0,184,136,184,160,184,0,0
760 '                                          [117]
770 ' --- Menue aufbauen                       [1540]
780 '                                          [117]
790 GOSUB 2430:POKE 30009,1:POKE 30010,12: [5208]
ENV 1,15,-1,10:ENV 2,15,-1,2
800 MODE 1:BORDER 0:INK 0,0:INK 1,14:INK 2 [6296]
,25:INK 3,17:WINDOW#2,25,38,1,24:WINDOW#3,
26,37,20,23
810 WINDOW#2,25,38,1,24:WINDOW#3,26,37,20, [2632]
23
820 PEN 3:PEN#1,3:PEN#2,0:PEN#3,0:PAPER#1, [4673]
0:PAPER#2,1:PAPER#3,1:CLS#2:CLS#3
830 LOCATE#2,4,2:PRINT#2,CHR$(225);CHR$(22 [7048]
6);CHR$(227);CHR$(228);CHR$(229):LOCATE#2,
7,3:PRINT#2,CHR$(221);CHR$(222);CHR$(223);
CHR$(224)
840 LOCATE#2,3,4:PRINT#2,CHR$(215);CHR$(21 [13072]
6);CHR$(207);CHR$(207);CHR$(207);CHR$(208)
;CHR$(207);CHR$(207);CHR$(212):LOCATE#2,3,
5:PRINT#2,CHR$(217);CHR$(218);CHR$(205);CH
R$(206);CHR$(203);CHR$(209);CHR$(210);CHR$
(205);CHR$(213)
850 LOCATE#2,3,6:PRINT#2,CHR$(219);CHR$(22 [10327]
0);CHR$(204);CHR$(204);CHR$(204);CHR$(204)
;CHR$(211);CHR$(204);CHR$(214):LOCATE#2,3,
```

```
7:PRINT#2,CHR$(164);CHR$(252);CHR$(253);CH
R$(193)
860 LOCATE#2,8,7:PRINT#2,CHR$(194);CHR$(19 [7332]
5);CHR$(196);CHR$(197);CHR$(198);CHR$(199)
:LOCATE#2,2,9:PRINT#2,CHR$(190);CHR$(191);
CHR$(192)
870 LOCATE#2,2,11:PRINT#2,CHR$(183);CHR$(1 [5923]
84);CHR$(185);CHR$(32);CHR$(188);CHR$(189)
:LOCATE#2,2,16:PRINT#2,CHR$(176);CHR$(177)
;CHR$(178)
880 LOCATE#2,2,17:PRINT#2,CHR$(179);CHR$(1 [6693]
80);CHR$(181);CHR$(182):LOCATE#2,6,16:PRIN
T#2,CHR$(58)
890 LOCATE#2,6,17:PRINT#2,CHR$(58):LOCATE# [5066]
2,9,11:PRINT#2,CHR$(58)
900 GRAPHICS PEN 2                         [948]
910 MOVE 390,280:DRAWR 208,0:DRAWR 0,-66:D [3850]
RAWR-208,0:DRAWR 0,66
920 MOVE 390,198:DRAWR 170,0:DRAWR 0,-80:D [2523]
RAWR-170,0:DRAWR 0,80
930 MOVE 390,102:DRAWR 208,0:DRAWR 0,-80:D [2914]
RAWR-208,0:DRAWR 0,80
940 GRAPHICS PEN 1                         [949]
950 '                                      [117]
960 ' --- Spielfeld aufbauen               [847]
970 '                                      [117]
980 GOTO 1950                              [341]
990 CLEAR:WINDOW#0,1,24,1,25:CLS#0:xm=PEEK [6878]
(30010):ym=xm:bm=INT((1/(11-2*PEEK(30009))
)*(xm/2)^2)
1000 DIM kx(bm+1),ky(bm+1),d(xm,ym),bx(bm) [3011]
,by(bm)
1010 GOSUB 1830:PEN 1:FOR y=1 TO ym STEP 2 [4229]
:FOR x=1 TO xm STEP 2
1020 LOCATE x,y:PRINT CHR$(239);CHR$(240): [3029]
LOCATE x,y+1:PRINT CHR$(241);CHR$(242);:NE
XT x,y
1030 '                                     [117]
1040 ' --- Bombenkoordinaten festlegen     [1463]
1050 '                                     [117]
1060 RANDOMIZE TIME:FOR a=1 TO bm          [1678]
1070 x=2*INT(RND*(xm/2)+1)-1:y=2*INT(RND*( [4068]
ym/2)+1)-1:bx(a)=x:by(a)=y
1080 FOR s=1 TO a-1:IF x=bx(s)AND y=by(s)T [4593]
HEN 1070 ELSE NEXT:NEXT
1090 FOR a=1 TO bm-1:FOR b=a+1 TO bm:IF bx [3786]
(a)<=bx(b)GOTO 1110
1100 hi=bx(b):bx(b)=bx(a):bx(a)=hi:hi=by(b [5379]
):by(b)=by(a):by(a)=hi
1110 NEXT b,a:GOTO 1320                    [642]
1120 '                                     [117]
1130 ' --- Anzahl der Bomben neben einem F [3837]
eld ausrechnen
1140 '                                     [117]
1150 IF d(x,y)>0 THEN GRAPHICS PEN 2:RETUR [1974]
N
1160 GOSUB 1790:FOR a=1 TO z:IF kx(a)=x AN [4214]
D ky(a)=y THEN RETURN ELSE NEXT
1170 FOR a=1 TO bm:IF x=bx(a)AND y=by(a)TH [6653]
EN GOSUB 1540:CLEAR INPUT:GOTO 2000:ELSE N
EXT:GOSUB 1190:RETURN
1180 IF x=bx(a)AND y=by(a)THEN GOSUB 1540: [6244]
CLEAR INPUT:GOTO 2000:ELSE NEXT:GOSUB 1190
:RETURN
1190 FOR a=1 TO bm:IF bx(a)=x-2 THEN low=a [4332]
:GOTO 1220:ELSE NEXT
1200 FOR a=1 TO bm:IF bx(a)=x THEN low=a:G [3091]
OTO 1220:ELSE NEXT
1210 FOR a=1 TO bm:IF bx(a)=x+2 THEN low=a [3654]
 ELSE NEXT
1220 FOR a=bm TO 1 STEP-1:IF bx(a)=x+2 THE [6160]
N high=a:GOTO 1250:ELSE NEXT
1230 FOR a=bm TO 1 STEP-1:IF bx(a)=x THEN  [5131]
high=a:GOTO 1250:ELSE NEXT
1240 FOR a=bm TO 1 STEP-1:IF bx(a)=x-2 THE [4574]
N high=a:ELSE NEXT
1250 FOR a=low TO high:IF by(a)=y+2 OR by( [5490]
a)=y OR by(a)=y-2 THEN d=d+1
1260 NEXT:low=0:high=0                     [781]
1270 IF d=0 THEN PRINT#1,CHR$(251);CHR$(25 [5083]
1);CHR$(251);CHR$(251);:d(x,y)=1:GOSUB 153
0:RETURN
1280 PEN#1,1:PRINT#1,CHR$(230+d);CHR$(251) [6154]
;CHR$(251);CHR$(251);:d(x,y)=1:d=0:GOSUB 1
530:PEN#1,3:RETURN
1290 '                                     [117]
1300 ' --- Zeit anzeigen                   [926]
1310 '                                     [117]
1320 x=1:y=1:WINDOW#1,x,x+1,y,y+1:GRAPHICS [5487]
 PEN 2:GOSUB 1530:ti=TIME
1330 cl=INT((TIME-ti)/300):mi=INT(cl/60):s [3970]
ec=cl-(mi*60)
1340 IF mi<10 THEN LOCATE#2,8,9 ELSE LOCAT [3626]
E#2,7,9
1350 PRINT#2,mi                            [381]
1360 IF sec<10 THEN LOCATE#2,11,9:PRINT#2, [5219]
CHR$(48)+CHR$(sec+48):ELSE LOCATE#2,10,9:P
RINT#2,sec
1370 '                                     [117]
1380 ' --- Tastaturabfrage                 [2297]
1390 '                                     [117]
1400 t$=UPPER$(INKEY$):IF t$=""THEN 1330 E [7008]
LSE t=ASC(t$):IF t<240 OR t>244 THEN:IF t<
>224 THEN:IF t$<>" "THEN:IF t<>13 THEN:IF
t<>127 THEN 1330
1410 IF t=240 AND y>2 THEN GRAPHICS PEN 0: [3686]
GOSUB 1530:y=y-2:GOSUB 1530
1420 IF t=241 AND y<ym-2 THEN GRAPHICS PEN [2228]
 0:GOSUB 1530:y=y+2:GOSUB 1530
1430 IF t=242 AND x>2 THEN GRAPHICS PEN 0: [3295]
GOSUB 1530:x=x-2:GOSUB 1530
1440 IF t=243 AND x<xm-2 THEN GRAPHICS PEN [3490]
 0:GOSUB 1530:x=x+2:GOSUB 1530
1450 IF t=224 THEN GRAPHICS PEN 2:GOSUB 15 [3109]
30:GOSUB 1150
1460 IF t=127 THEN GRAPHICS PEN 0:GOSUB 15 [3345]
30:GOSUB 1570:GOTO 2000
1470 IF t$=" "THEN GRAPHICS PEN 1:GOSUB 16 [4179]
30:GOSUB 1530
1480 IF t=13 THEN GOSUB 1870               [1034]
1490 WINDOW#1,x,x+1,y,y+1:GOTO 1330        [2529]
1500 '                                     [117]
1510 ' --- Rahmencursor zeichnen           [1045]
1520 '                                     [117]
1530 MOVE(2*x-2)*8,398-(2*y-2)*8:DRAWR 0,- [5829]
30:DRAWR 30,0:DRAWR 0,30:DRAWR-30,0:GRAPHI
CS PEN 2:RETURN
1540 '                                     [117]
1550 ' --- Game over + alle bomben zeigen  [2146]
1560 '                                     [117]
1570 GOSUB 1780                            [855]
1580 PEN 2:FOR a=1 TO bm:LOCATE bx(a),by(a [4557]
):PRINT CHR$(243);CHR$(244)
1590 LOCATE bx(a),by(a)+1:PRINT CHR$(245); [4938]
CHR$(246):NEXT:PEN 1:RETURN
1600 '                                     [117]
1610 ' --- Bomben mit Kreuz kennzeichnen   [3358]
1620 '                                     [117]
1630 IF d(x,y)>0 THEN GRAPHICS PEN 2:RETUR [1974]
N
1640 IF bm=z THEN:FOR a=1 TO z ELSE 1660   [3385]
1650 IF kx(a)=x AND ky(a)=y THEN 1680 ELSE [4166]
 NEXT a:GRAPHICS PEN 2:RETURN
1660 z=z+1:FOR a=1 TO z:CLS#1:IF kx(a)=x A [3053]
ND ky(a)=y THEN 1680 ELSE NEXT a
1670 GOSUB 1770:PEN#1,3:PRINT#1,CHR$(247); [6620]
CHR$(248);CHR$(249);CHR$(250);:kx(z)=x:ky(
z)=y:GOSUB 1830:GRAPHICS PEN 2:RETURN
1680 GOSUB 1770:PEN#1,1:PRINT#1,CHR$(239); [6753]
CHR$(240);CHR$(241);CHR$(242);:kx(a)=0:l=l
+1:PEN#1,3:GRAPHICS PEN 2:ELSE RETURN
1690 FOR q=1 TO l:FOR s=1 TO z             [1243]
1700 IF kx(s)=0 THEN FOR a=s TO z:kx(a)=kx [1894]
(a+1):ky(a)=ky(a+1):NEXT a
1710 NEXT s,q:g=z:z=0:FOR a=1 TO g:IF kx(a [3769]
)>0 THEN z=z+1
1720 NEXT a:q=0:GOSUB 1830:RETURN          [1287]
1730 '                                     [117]
1740 ' --- Sound                           [877]
1750 '                                     [117]
1760 SOUND 1,500,15,15:SOUND 1,400,15,15:S [3873]
OUND 1,300,15,15:SOUND 1,200,15,15:RETURN
1770 SOUND 1,120,5,15:SOUND 1,100,5,15:SOU [3222]
ND 1,80,5,15:SOUND 1,60,5,15:RETURN
1780 SOUND 1,900,0,15,1,,15:RETURN         [2234]
1790 SOUND 1,650,0,15,2:RETURN             [2322]
1800 '                                     [117]
1810 ' --- Anzahl der Kreuze anzeigen      [2247]
1820 '                                     [117]
1830 IF bm-z<10 THEN LOCATE#2,11,11:PRINT# [6336]
2,bm-z;:RETURN ELSE LOCATE#2,10,11:PRINT#2
,bm-z;:RETURN
1840 '                                     [117]
1850 ' --- Kontolle ob Kreuze auch wirklic [4860]
h Bomben sind
1860 '                                     [117]
1870 IF z<>bm THEN RETURN                  [1014]
1880 GRAPHICS PEN 0:GOSUB 1530:FOR a=1 TO  [3324]
z:FOR b=1 TO z
1890 IF bx(a)=kx(b)AND by(a)=ky(b)THEN v=v [1700]
+1
1900 NEXT b,a:m=5-PEEK(30009):hi=(PEEK(299 [5019]
```

```
98+(2*m)))*255+PEEK(29998+(2*m)+1)
1910 IF v=z AND cl<=hi THEN GOSUB 1760:GOS [6845]
UB 1580:FOR a=1 TO 400:NEXT:GOSUB 2430:GOS
UB 2510 ELSE GOSUB 1570:GOTO 2000
1920 '                                     [117]
1930 ' --- auswahlmenue                    [1845]
1940 '                                     [117]
1950 GOSUB 2190:CLS#3                      [1368]
1960 LOCATE#3,3,1:PRINT#3,CHR$(170);CHR$(1 [5787]
71);CHR$(172);CHR$(173);CHR$(174);CHR$(175
)
1970 LOCATE#3,3,2:PRINT#3,CHR$(162);CHR$(1 [4987]
63);CHR$(165);CHR$(166);CHR$(167);CHR$(168
);CHR$(169)
1980 LOCATE#3,3,3:PRINT#3,CHR$(157);CHR$(1 [3872]
58);CHR$(159);CHR$(160);CHR$(161)
1990 LOCATE#3,3,4:PRINT#3,CHR$(139);CHR$(1 [2788]
40);CHR$(141);CHR$(142);CHR$(143)
2000 y=1:LOCATE#3,1,1:PRINT#3,CHR$(62):LOC [3865]
ATE#3,12,1:PRINT#3,CHR$(60);
2010 t$=UPPER$(INKEY$):IF t$=""THEN 2010 E [5333]
LSE t=ASC(t$):IF t<240 OR t>241 THEN:IF t<
>224 THEN 2010
2020 GOSUB 2070:IF t=224 THEN GOSUB 1790:E [2217]
LSE 2010
2030 IF y=1 THEN z=0:GRAPHICS PEN 1:PEN 1: [7946]
LOCATE#3,1,1:PRINT#3,CHR$(32);:LOCATE#3,12
,1:PRINT#3,CHR$(32);:GOTO 990
2040 IF y=2 THEN 2100                      [944]
2050 IF y=3 THEN GOSUB 2310:GOSUB 2470:GOT [2277]
O 2010
2060 IF y=4 THEN END                       [632]
2070 IF t=240 AND y>1 THEN LOCATE#3,1,y:PR [7648]
INT#3,CHR$(32):LOCATE#3,12,y:PRINT#3,CHR$(
32);:y=y-1:LOCATE#3,1,y:PRINT#3,CHR$(62):L
OCATE#3,12,y:PRINT#3,CHR$(60);:RETURN
2080 IF t=241 AND y<4 THEN LOCATE#3,1,y:PR [8283]
INT#3,CHR$(32):LOCATE#3,12,y:PRINT#3,CHR$(
32);:y=y+1:LOCATE#3,1,y:PRINT#3,CHR$(62):L
OCATE#3,12,y:PRINT#3,CHR$(60);:RETURN
2090 RETURN                                [555]
2100 CLS#3:LOCATE#3,3,1:PRINT#3,CHR$(138); [5398]
CHR$(137);CHR$(136);CHR$(135);CHR$(230)
2110 LOCATE#3,3,2:PRINT#3,CHR$(153);CHR$(1 [2372]
54);CHR$(155);CHR$(156)
2120 LOCATE#3,3,3:PRINT#3,CHR$(150);CHR$(1 [2684]
51);CHR$(152)
2130 LOCATE#3,3,4:PRINT#3,CHR$(144);CHR$(1 [4586]
45);CHR$(146);CHR$(147);CHR$(148);CHR$(149
)
2140 y=1:LOCATE#3,1,1:PRINT#3,CHR$(62):LOC [3865]
ATE#3,12,1:PRINT#3,CHR$(60);
2150 t$=UPPER$(INKEY$):IF t$=""THEN 2150 E [5435]
LSE t=ASC(t$):IF t<240 OR t>241 THEN:IF t<
>224 THEN 2150
2160 GOSUB 2070                            [869]
2170 IF t=224 THEN POKE 30009,y:POKE 30010 [5008]
,4*(y+2):GOSUB 1790:GOSUB 2190:GOTO 1950
2180 GOTO 2150                             [349]
2190 IF PEEK(30009)=1 THEN l$=CHR$(138)+CH [5413]
R$(137)+CHR$(136)+CHR$(135)+CHR$(230)+CHR$
(32)
2200 IF PEEK(30009)=2 THEN l$=CHR$(153)+CH [4627]
R$(154)+CHR$(155)+CHR$(156)+CHR$(32)+CHR$(
32)
2210 IF PEEK(30009)=3 THEN l$=CHR$(150)+CH [3664]
R$(151)+CHR$(152)+CHR$(32)+CHR$(32)+CHR$(3
2)
2220 IF PEEK(30009)=4 THEN l$=CHR$(144)+CH [4036]
R$(145)+CHR$(146)+CHR$(147)+CHR$(148)+CHR$
(149)
2230 bm=PEEK(30009):xm=PEEK(30010):ym=xm:L [2619]
OCATE#2,2,14:PRINT#2,l$
2240 IF xm*ym/4>99 THEN LOCATE#2,7,16 ELSE [3188]
 LOCATE#2,8,16
2250 PRINT#2,xm*ym/4;:bm=INT((1/(11-2*PEEK [3542]
(30009)))*(xm/2)^2)
2260 IF bm<10 THEN LOCATE#2,9,17 ELSE LOCA [2458]
TE#2,8,17
2270 PRINT#2,bm;:RETURN                    [1157]
2280 '                                     [117]
2290 ' --- High-Score-Tabelle darstellen   [2980]

2300 '                                     [117]
2310 GRAPHICS PEN 1:WINDOW#0,1,24,1,25:CLS [2939]
#0
2320 MOVE 6,18:DRAWR 370,0:DRAWR 0,358:DRA [1598]
WR-98,0
2330 MOVE 6,18:DRAWR 0,358:DRAWR 78,0      [1706]
2340 MOVE 86,390:DRAWR 192,0:DRAWR 0,-28:D [3529]
RAWR-192,0:DRAWR 0,28
2350 PEN 2:LOCATE 7,2:PRINT"HI - SCORES":L [4062]
OCATE 20,3:PRINT"TIME";
2360 LOCATE 2,5:PRINT CHR$(144);CHR$(145); [4634]
CHR$(146);CHR$(147);CHR$(148);CHR$(149)
2370 LOCATE 2,10:PRINT CHR$(150);CHR$(151) [1890]
;CHR$(152)
2380 LOCATE 2,15:PRINT CHR$(153);CHR$(154) [4621]
;CHR$(155);CHR$(156)
2390 LOCATE 2,20:PRINT CHR$(138);CHR$(137) [4597]
;CHR$(136);CHR$(135);CHR$(230)
2400 '                                     [117]
2410 ' --- High-Score-Tabelle von Diskette [3570]
 lesen
2420 '                                     [117]
2430 PEN 1:OPENIN"think.sco":FOR a=4 TO 1  [4673]
STEP-1:FOR b=1 TO 4
2440 INPUT#9,name$(a,b):INPUT#9,s(a,b):NEX [3237]
T b,a:CLOSEIN
2450 FOR a=1 TO 4:POKE 29998+(2*a),INT(s(a [2850]
,4)/255)
2460 POKE 29998+(2*a)+1,s(a,4)-(INT(s(a,4) [2253]
/255)*255):NEXT:RETURN
2470 FOR a=1 TO 4:FOR b=1 TO 4:LOCATE 9,a* [3724]
5+b-1:PRINT name$(a,b)
2480 mi=INT(s(a,b)/60):sec=s(a,b)-(mi*60): [3560]
LOCATE 19,a*5+b-1:PRINT mi:LOCATE 22,a*5+b
-1
2490 IF sec<10 THEN PRINT CHR$(48)+CHR$(se [6380]
c+48)ELSE PRINT CHR$(48+INT(sec/10));CHR$(
48+sec-((INT(sec/10))*10));
2500 NEXT:NEXT:RETURN                      [1435]
2510 FOR a=1 TO 4:IF s(m,a)>=cl THEN 2520  [2673]
ELSE NEXT
2520 FOR b=4 TO a+1 STEP-1:s(m,b)=s(m,b-1) [4056]
:NEXT:s(m,a)=cl:GOSUB 2630
2530 '                                     [117]
2540 ' --- Highscore-Tabelle auf Diskette  [2956]
schreiben
2550 '                                     [117]
2560 OPENOUT"think.sco":FOR b=4 TO a+1 STE [2446]
P-1
2570 name$(m,b)=name$(m,b-1):NEXT:name$(m, [2456]
a)=a$:FOR a=4 TO 1 STEP-1
2580 FOR b=1 TO 4:PRINT#9,name$(a,b):PRINT [4160]
#9,s(a,b):NEXT:NEXT:CLOSEOUT
2590 GOSUB 2310:GOSUB 2450:GOSUB 2470:GOTO [2664]
 2000
2600 '                                     [117]
2610 ' --- Namen eingeben                  [1574]
2620 '                                     [117]
2630 WINDOW#0,1,23,1,25:CLS#0              [1775]
2640 MOVE 2,370:DRAWR 372,0:DRAWR 0,-150:D [3736]
RAWR-372,0:DRAWR 0,150
2650 MOVE 62,300:DRAWR 180,0:DRAWR 0,-42:D [3343]
RAWR-180,0:DRAWR 0,42
2660 LOCATE 2,4:PRINT"Please enter your Na [3203]
me";
2670 WINDOW#4,6,14,8,8:PAPER#4,0:PEN#4,1:C [2620]
LS#4
2680 t$=UPPER$(INKEY$):IF t$=""THEN 2680 E [6556]
LSE t=ASC(t$):SOUND 1,400,6,10:IF t<65 OR
t>90 THEN:IF t<>13 THEN:IF t<>127 THEN 268
0
2690 IF t=13 THEN:IF LEN(a$)=0 THEN 2680 E [1865]
LSE RETURN
2700 IF LEN(a$)=0 AND t=127 THEN CLS#4:a$= [3058]
"":GOTO 2680
2710 IF LEN(a$)=9 AND t<>127 THEN 2680     [1665]
2720 IF t=127 THEN LOCATE#4,LEN(a$),1:PRIN [8117]
T#4,CHR$(32);:a$=LEFT$(a$,LEN(a$)-1):LOCAT
E#4,1,1:PRINT#4,a$;:ELSE a$=a$+t$:LOCATE#4
,1,1:PRINT#4,a$;
2730 GOTO 2680                             [387]
```

```
100 ' THINK-SC.BAS 'Generiert Highscore-Ta [2731]
belle
110 ' CPC 664/6128(plus)                   [1532]
120 ' (c) 1992 Ivo Mieskes                 [1279]
130 '          & CPC International         [2118]
140 OPENOUT "think.sco":DIM name$(4,4):FOR [2851]
 a=1 TO 4:FOR b=1 TO 4
150 name$(a,b)="CPC INTERNATIONAL":NEXT b, [1924]
a
160 FOR a=1 TO 4:s(1,a)=a*30:s(2,a)=60+a*3 [3783]
0:s(3,a)=180+a*30:s(4,a)=420+a*30:NEXT
170 s(1,a)=a*30:s(2,a)=60+a*30:s(3,a)=180+ [3055]
a*30:s(4,a)=420+a*30
180 FOR a=1 TO 4:FOR b=1 TO 4              [1551]
190 PRINT #9,name$(a,b):PRINT #9,s(a,b):NE [5342]
XT:NEXT:CLOSEOUT
```

... und wo lassen Sie rechnen?

"Adam Riese" – eine leistungsfähige Tabellenkalkulation auf der DATABOX

Wo es um Zahlen geht, ist der Computer in seinem Element. Rechnen, das kann er. Dennoch gibt es gerade für die CPCs nur wenige Programme, die Zahlenwerte übersichtlich, flexibel und zugleich ohne lästige Umwege verarbeiten können. "Adam Riese" ist ein solches Programm –, ein Helfer für alle Rechenfälle, ob Sie nun ein Haushaltsbuch führen, Benzinverbräuche verfolgen, Kontobewegungen dokumentieren oder Lagerbestände verwalten wollen.

Diesmal haben wir mit der Tabellenkalkulation "Adam Riese" wirklich ein besonders leistungsfähiges Anwendungsprogramm für Sie aufgestöbert. "Adam Riese" läuft auf allen CPCs – ob 464, 664, 6128 oder Plus – und läßt sich notfalls sogar auf Kassette einsetzen. Eine ausgeklügelte Installationsroutine erkennt Ihr CPC-Modell automatisch und paßt den Maschinenteil des Programms entsprechend an. Sie finden "Adam Riese" als Bonusprogramm auf der DATABOX-Diskette zu dieser Heftausgabe.

Flott installiert

Bevor der "Rechenknecht" seine Arbeit aufnehmen kann, wird mit dem Installationsprogramm "INSTADAM" ein passendes Binärfile für 464, 664 oder 6128 erzeugt. Nach dem Start von "ADAM" meldet sich das Kalkulationsprogramm dann zunächst mit einer leeren Tabelle, der Cursor steht in der ersten Spalte ganz oben. Dies ist der "Zeigermodus"; hier stehen die folgende Befehle zur Verfügung.

<COPY>: Eingabe eines Werts an der Cursorposition (Abschließen mit <RETURN>

<CTRL>-<E>: Eingabe einer Wertegruppe in eine Spalte (Zeilenwechsel mit <RETURN>, abschließen mit erneutem <CTRL>-<E>)

<CTRL>-<R>: Auslösen des Rechenvorgangs; der gesamte Datenbereich wird durchgerechnet

<CTRL>-<M>: zum Menü

Das Menü enthält 16 Optionen. Das Ändern von Formel, Format und Spaltenbezeichnung bezieht sich ebenso wie "Formel ausgeben" jeweils auf die gesamte aktuelle Tabellenspalte. Der Cursor muß vor Aufruf des Menüs im "Zeigermodus" auf ein Feld der gewünschten Spalte gesetzt werden.

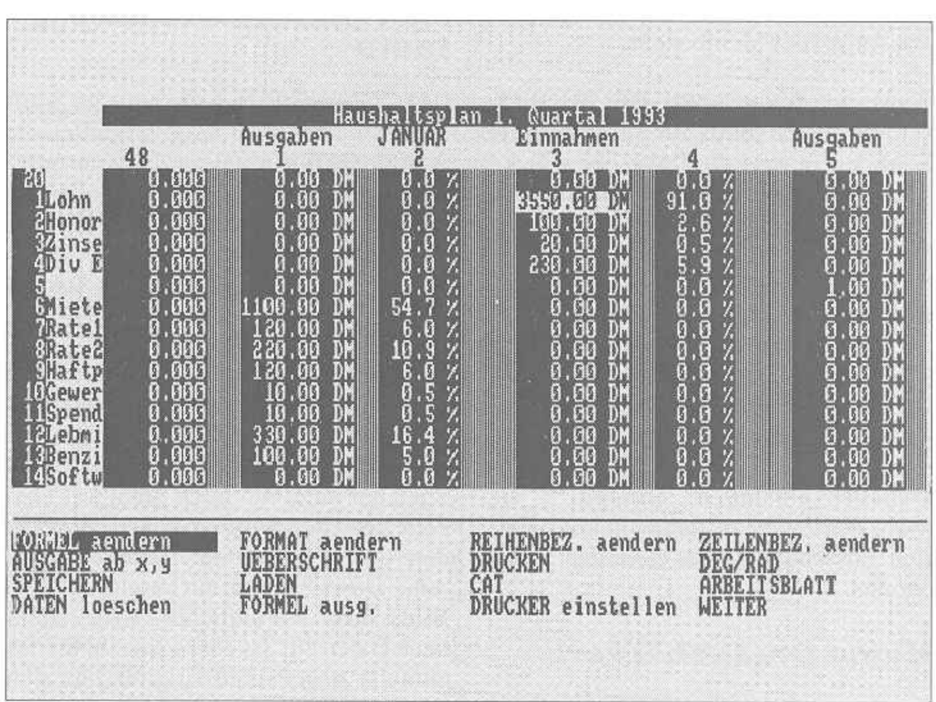
Mammut-Menü

Im Menü wird eine gewünschte Option mit den Cursortasten angewählt und mit <Return> bestätigt. Die Wahl des Menüpunkts <Weiter> führt zurück in den "Zeigermodus". Hier die Bedeutung der übrigen Menüpunkte:

Formel ändern: bestimmt, was in jedem Feld der gewählten Spalte geschehen soll. Voreinstellung ist "Rn", wobei statt n die Spaltennummer erscheint. Dies bedeutet, daß die Felder der Spalte n ihre Werte per Direkteingabe erhalten sollen.
Statt dessen läßt sich die gewählte Spalte auch anweisen, berechnete Werte darzustellen. Die Formel dafür wird wie vom Basic gewohnt formuliert, also etwa "R1+R2" (addiere die beiden Werte der gleichen Zeile, die in der 1. und 2. Spalte stehen) oder auch "2*SQR((R1)^3+1)-SIN(R1)".

Gib's ihm spaltenweise!

Über die arithmetischen Funktionen des Basic hinaus lassen sich in den Formeln für "Adam Riese" noch die beiden Variablen SUn (n=Spaltennummer) und ZE verwenden. ZE entspricht der Zeilennummer des jeweiligen Feldes. "2*ZE" erzeugt somit eine Spalte von fortlaufend in Zweier-Abstand numerierten Feldern. SUn steht für die Spaltensumme der angegebenen Spalte. Die Formel "R1*100/SU1" rechnet, wenn man sie für die 2. Spalte einträgt, alle Werte der 1. Spalte in Prozentwerte um.



Das Bildschirmfenster von "Adam Riese" zeigt immer nur einen Ausschnitt aus der jeweiligen Gesamttabelle und läßt sich mit den Cursortasten auf dieser verschieben. Die Größe der ganzen Tabelle ist im Rahmen des begrenzten CPC-Arbeitsspeichers frei einstellbar.

Achtung: In einem solchen Fall muß die 1. Spalte mindestens einen Wert enthalten, bevor man die Tabelle durchrechnen läßt. Anderenfalls wäre die Formel in Spalte 2 gezwungen, durch Null zu teilen, was mit einer Fehlermeldung bestraft würde. Bei einer Nulldivision ersetzt das Programm nach der Fehlermeldung den Nullbetrag intern durch den Wert 0,0000001, um weiterrechnen zu können.

Formel anzeigen: gibt über die Formel Auskunft, die der gewählten Spalte aktuell zugewiesen ist.

Format ändern: legt Stellenzahl, Kommaposition und Textergänzungen (Einheiten, Prozentzeichen) für die gewählte Spalte fest. Die Schreibweise entspricht der Syntax für den "PRINT USING()"-Befehl in Basic. Beispiel: Das Format "####.## DM" bringt bei Eingabe von "3" die Darstellung "3.00 DM". Die Gesamtlänge des Formats ist auf 12 Zeichen begrenzt. Nach Festlegung des Formats möchte das Programm noch wissen, ob es die Spaltensumme der gewählten Spalte in gleichem Format daruntersetzen soll oder nicht.

Reihen- und Zeilenbezeichnung ändern: Spaltenüberschriften dürfen höchstens zehn, Zeilentexte höchstens fünf Zeichen lang sein. Diese Befehle beziehen sich immer auf die Spalte beziehungsweise Zeile, in der der Cursor vor Aufruf des Menüs steht.

Ausgabe ab x,y: verschiebt das Bildschirmfenster so, daß das angegebene Feld der Tabelle die linke obere Ecke bildet. So ist bei größeren Tabellen ein schnelles Springen ohne lange Cursor-Wanderwege möglich.

Überschrift: Die Eingabe der Gesamtüberschrift für die Tabelle wird mit <Return> abgeschlossen, das Programm zentriert diesen bis zu 70 Zeichen langen Text automatisch.

Drucker einstellen: ermöglicht die Wahl des Zeilenabstands und das Ein- und Ausschalten eines etwaigen NLQ-Modus.

Keine Druckeberger!

Die Steuerzeichen im Programm sind auf einen DMP-2000-Matrixdrucker angelegt und sollten für alle Epson-FX-80-kompatiblen Geräte passen. Eine eventuelle Anpassung an andere Drucker ist kinderleicht, da der entsprechende Programmteil in Basic geschrieben und gut kommentiert ist.

Drucken: Hier ist zwischen 80- und 132-Zeichen-Modus zu wählen. Der erste bringt nur die auf dem Bildschirm sichtbaren 6 Spalten auf den Drucker; der 132-Zeichen-Modus schafft 10 Spalten. Dabei bestimmt die linke obere Ecke des Bildschirmfensters den Beginn des Ausdrucks.

DEG/RAD: bestimmt, ob in Formeln auftauchende Winkelfunktionen (SIN, COS, TAN, ATN) die ihnen übergebenen Werte als Gradzahlen (DEG) oder Bogenmaß (RAD) auffassen. Voreinstellung ist Bogenmaß.

Speichern: Das Programm bietet vier Möglichkeiten des Speicherns. *"Komplett"* speichert das Arbeitsblatt (Formeln, Formate, Überschriften usw.) und alle Daten ab. *"Daten"* beschränkt sich auf die Daten in den Feldern und erlaubt so den Export des Inhalts einer Tabelle in eine andere. *"Arbeitsblatt"* speichert sozusagen nur die Form und vergißt den Inhalt. So läßt sich etwa aus einer "durchgestylten" Buchführungs-Tabelle eine fertig eingerichtete Tabelle mit sämtlichen Formeln, aber leeren Wertefeldern fürs neue Jahr gewinnen.

Partner für Datenimporteure

"ASCII" speichert die jeweils im Bildschirmfenster sichtbaren 6 Spalten als ASCII-Textinformation und ermöglicht damit den Export von Tabellendaten zu einem ASCII-fähigen Textverarbeitungsprogramm (etwa dem "Text-Assistenten" von der letzten DATA-BOX!).

Laden: Analog zum Speichern gibt es hier drei Möglichkeiten. *"Komplett"* lädt eine mit "Komplett speichern" erzeugte Datei. Alle Daten, die gerade im Speicher sind und noch nicht auf Diskette gesichert wurden, gehen beim Laden einer "Komplett"-Tabelle verloren. *"Daten"* lädt in ein bereits auf dem Bildschirm befindliches Arbeitsblatt neue Daten ein. Ist deren Anzahl für das aktuelle Arbeitsblatt zu groß, läßt sich das Laden auf einen passenden Teil beschränken. *"Arbeitsblatt"* lädt ein Tabellenformular mit Formeln und Texten, dessen Wertefelder leer sind. Befindet sich bereits eine Tabelle im Speicher, so werden deren Daten in das geladene Arbeitsblatt übernommen. Dessen Größe läßt sich zu diesem Zweck an den vorhandenen Datenumfang anpassen. Verzichtet man auf die Anpassung eines zu großen Arbeitsblatts, werden die im Speicher befindlichen Daten gelöscht.

Cat: gibt ein Inhaltsverzeichnis der aktuellen Diskette im Bildschirmfenster aus

Arbeitsblatt: bestimmt die Dimensionen des Tabellenformulars. *"Größe verändern"* erhält dabei den Formel- und Überschriftenbestand sowie die bislang eingegebenen Werte. Allerdings läßt sich eine vorhandene Tabelle hiermit nur verkleinern, nicht aber erweitern. Beim Verkleinern wird kein Speicherplatz frei; die "abgeschnittenen" Bereiche der Tabelle lassen sich durch erneuten Aufruf dieses Menüpunkts wieder "ankleben". *"Neues Formblatt"* verwirft alle eventuell bereits im Speicher befindlichen Formeln und Daten. Es wird eine ganz jungfräuliche Tabelle mit frei bestimmbarer Größe zur Verfügung gestellt.

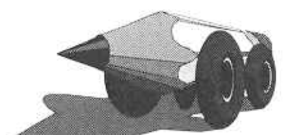
Daten löschen: erhält die Formeln und Überschriften einer im Speicher befindlichen Tabelle, setzt jedoch alle eingegebenen Werte auf Null.

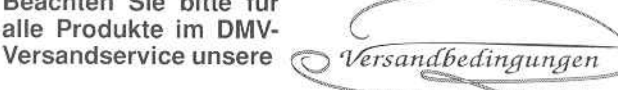
Rechnen mit Spaß

Was die Geschwindigkeit der Werteverarbeitung angeht, wird das Programm zwar keine Rekorde brechen. Da es beim Rechnen aber weitgehend auf Maschinenspracheroutinen zurückgreift, erreicht es dennoch ein akzeptables Arbeitstempo. Freilich können allzu große Tabellenumfänge und komplexe Formeln, besonders Wurzel- oder trigonometrische Funktionen, die Sache dramatisch verzögern. Es empfiehlt sich, nur zeitweise benötigte Formelbereiche am Tabellenrand zu plazieren, um sie bei Bedarf durch Ändern der aktiven Tabellengröße abzuschalten.

"Adam Riese" ist zwar ein Programm für harte Zahlenbeißerei und nicht zum Spielen. Wir sind jedoch überzeugt, daß die Arbeit mit unserem komfortablen Rechengenie auch Ihnen Spaß machen wird.

(sz)





Ihre Quelle für bewährte Qualitätssoftware: Bei uns finden Sie Futter für Ihren Computer — zum Mini-Preis!

Beachten Sie bitte für alle Produkte im DMV-Versandservice unsere *Versandbedingungen*

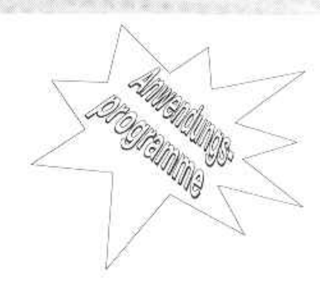


*Unabhängig von der Anzahl der bestellten Programme oder Hefte berechnen wir bei jeder Sendung für das Inland 4,— DM, für das Ausland 6,— DM Porto und Verpackung. Alle Software-Produkte für CPC und Joyce werden, wenn nicht anders angegeben, auf 3-Zoll-Disketten ausgeliefert. Bitte beachten Sie die jeweiligen Bestellnummern und benutzen Sie die diesem Heft beigefügten Bestellkarten.

DMV-Verlag • Postfach 250 • 3440 Eschwege



für CPC 464, 664 und 6128:



CopyShop

Das universelle Bildschirmdruck-Programm für alle CPCs. Wollen Sie Ihre traumhaft schöne Fraktalgrafik im Freundeskreis zeigen, oder brauchen Sie zu Ihrem selbstgeschriebenen Programm einen Kontroll-Ausdruck? CopyShop bringt Ihre Bildschirmansichten zu Papier, auch mit den exotischsten Druckern. Verschiedene Schraffuren für Grautöne lassen sich den einzelnen Bildschirmfarben frei zuordnen. Ein Show-Modus zeigt Ihnen die Hardcopy vor dem endgültigen Ausdruck auf dem Bildschirm!

- Verarbeitet Bilder im Standard-Screenformat (17 kByte-Binärfiles)
- Vier Ausdruck-Formate von handlich bis DIN A 4 quer
- Arbeitet in allen drei CPC-Bildschirmmodi
- Integrierter Grafikeditor mit Füllfunktion und Druckvoraussicht
- Beliebige Ausschnittsvergrößerungen und Invertierung
- Anpassungsmenü für alle Epson- und Seikosha-Drucker
- Auch für gedrehte Bitbild-Bytes, z.B. beim NEC P2
- Mit speicherplatzsparendem Screendump-Utility zum Vorladen — erzeugt auf Tastendruck Bilddateien im Standard-17-kByte-Format aus beliebigen laufenden Programmen heraus
- Mit selbstrelozierender Hardcopy-Routine für eigene Programme

Best.-Nr. 202, DM 49,-

Turbo-DATA-CPC

Volldampf in der Dateiverwaltung

Brauchen Sie ein neues Adreßbuch oder Telefonverzeichnis? Müssen Sie Ihre Sammlungen von Briefmarken, Schallplatten, Dias oder anderen Schätzen sortieren? Alles kein Problem mit **Turbo-DATA-CPC**, der universellen Datenbank für alle CPCs mit Diskettenlaufwerk.

Turbo-DATA-CPC hilft Ihnen, Übersicht und Ordnung in Ihre Daten zu bringen. Ein kurzer Blick auf einige Leistungsmerkmale gefällig? — Bitte sehr:

- Universell durch veränderbare Ein- und Ausgabemasken
- Deutsche Umlaute mit DIN-Tastaturbelegung
- Blitzschnelle Suchfunktionen durch Indexfelder
- Dateikapazität bis zu 80 verschiedenen Feldern
- Bis zu 19 Felder gleichzeitig auf dem Bildschirm sichtbar
- Besonders schnelle und umfangreiche Sortierfunktion
- Eigene Disketten-Formatierroutine mit besonders hoher Kapazität
- Zweiteilung des Bildschirms in Status- und Arbeitsfeld
- Komplette Druckroutinen, auch für Etikettendruck

Best.-Nr. 214, DM 69,-*

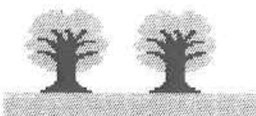
Für Profis: CPC Special Offer 2

Vier Programme der Star-Division im Sonderpreis-Paket

Disksort-Star: Leistungsstarke Diskettenverwaltung mit hohem Bedienungskomfort und integriertem Diskettenmanager. Verwalten, Archivieren, Katalogisieren Ihrer Disketten und Ausdruck der Daten — mit Disksort-Star kein Problem.

Designer-Star: Malprogramm zum komfortablen Anfertigen von Bildschirmgrafiken. Zur Bedienung genügt die vertraute Tastatur. Auf Tastendruck werden Sie per Hilfsmenü durch das Programm geführt.

Creator-Star: Machen Sie Ihre eigenen CPC-Basic-Programme durch Trickfilmsequenzen mit animierter Zeichensatzgrafik interessanter. Der Creator-Star bietet je einen Editor für stehende und animierte Motive (Kulissen- und Shape-Designer). Die Editoren sind als Programmgeneratoren ausgelegt und erzeugen Basic-Code, der innerhalb Ihrer eigenen Programme weiterverarbeitet und eingesetzt werden will. Beliebige

Laufschrift-Texte lassen sich integrieren. In einer eigenen Programmiersprache lassen sich Shapes miteinander verbinden und Kulissen übereinanderlegen.

Star-Mon: Professionelle CPC-Programmierung mit Assembler, Disassembler und Monitor. Das System besitzt einen eigenen Diskettenmonitor und Editor, eine Trace-Funktion, kann Breakpoints setzen und Bankswitching vornehmen.

4 Programme für CPC-Profis im Paket: Best.-Nr. 205, DM 59,-*

ConText CPC: die alltagstaugliche Textverarbeitung für jeden CPC!

```
Testtext für Context CPC
▸Immer wieder schön und naß wie Hw2w0▸

Dies  ist  ein  Probetext für Context CPC . Ich glaube, wir werden eine Menge
Spaß miteinander haben. Dieses Programm macht ja schon ziemlich viel von dem,
was man von einer Textverarbeitung erwartet.

Was halten wir denn von diesem hübschen Pseudo-WYSIWYG? Das gleiche haben wir
auf  dem  8-Bit-Atari  ja auch schon gesehen, allerdings noch schöner und mit
Invertierung für Kursiv-Passagen. Außerdem konnte man bei Austro-Text bereits
die Vorteile eines Online-Umbruchs genießen...

Warum  macht  man  sich's nicht leicht und legt sich auf
die  nächste  Wiese, anstatt im muffigen Zimmer die Zeit
mit Arbeit zu verbringen?

Dies ist ein Test-Text, wie jedermann weiß.
Dies ist   e c h t   breit...

Achten  Sie  bitte  auf  die
Umbruchs-Schlaglöcher...
```

Lassen Sie Ihren CPC nicht verstauben! Mit ConText CPC erwerben Sie ein Anwendungsprogramm, das eine überzeugende Leistung bei einfacher Bedienung bietet. Die ausführliche deutsche Anleitung macht es auch dem technischen Laien möglich, schon nach kürzester Zeit Briefe, Referate und vieles andere mit dem Computer zu schreiben.

Seit seinem Erscheinen 1986 ist dieses Schreibprogramm für den CPC viele tausend Mal verkauft worden. Die Umsetzung auf den PC fand inzwischen sogar sechsstellige Benutzerzahlen. Kein Wunder: ConText ist ein simples Programm, ohne Schnörkel und unnötige Kompliziertheiten. Man kommt schnell damit klar und erreicht das Ziel — die tägliche Schreibarbeit zu vereinfachen — ohne Umwege. Es ist ein Programm, das in erster Linie nicht den "Freak" und Spezialisten, sondern den Praktiker anspricht. Den Studenten, der seine Referate ohne Tippfehler abgeben möchte. Den Pfarrer, der seine Predigtentwürfe sauber und wiederverwendbar abspeichern und archivieren möchte. Den Vereinsvorstand, der das neue Jahresprogramm aushangsreif zu Papier bringen will. Und natürlich den Programmierer, der zu seinem Werk eine ansprechende und gut lesbare Dokumentation liefern möchte. Also: Man könnte sagen, daß ConText so etwas wie der "Käfer" unter den Textprogrammen ist — es läuft und läuft und läuft...

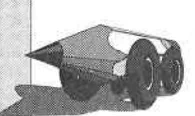
Daß die Darstellung der Texte darüber hinaus auch noch besonders ansprechend ist und alle Programmfunktionen erfreulich flott arbeiten, ist das Verdienst des Programmautors Matthias Uphoff, der sich auch durch andere Projekte ("Copy-Shop", "das Software-Experiment") einen Namen in der CPC-Szene gemacht hat.

Mit ConText CPC macht das Schreiben richtig Spaß:

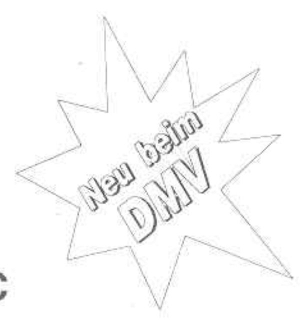
- Einfügen, Fließtext, Blockformatierung und Kopieren auf Tastendruck
- 25 kByte Textspeicher, ausreichend für mindestens fünf DIN-A4-Seiten
- Mehrspaltige Texte und Kopieren für besondere Aufgaben
- Textblöcke von Diskette oder Kassette nachladbar und jederzeit im Text zu plazieren
- Drucken im Hintergrund, während Sie weiter an Ihren Texten arbeiten
- Echtbild-Darstellung der Schriftarten Vergrößert und Unterstrichen
- Ins Textprogramm integrierter Taschenrechner und Kalender
- Variable Druckeranpassung mit Einstellmöglichkeit in Tabellenform
- WordStar-kompatible Tastaturkommandos für den leichten Umstieg

Best.-Nr. 207, DM 59,-*

CPC Joyce Preis-Offensive



Die Super-Serie



Das Beste von RE-Software: Ausgewählte und geprüfte Qualitäts-Software für CPC

Jetzt exklusiv zum Lächel-Preis *beim DMV*

Wie wär's denn damit:

The Player's Dream Vol. 3

Eine Sammlung besonders attraktiver Spiele, professionell programmiert und mit detailreicher Grafik ausgestattet. **Best.-Nr. 194, DM 19,-***

Ritter Kunibert: Eines der witzigsten Reaktionsspiele für den CPC, tempogeladen und mit detailreicher Farbgrafik. Verhelfen Sie Kunibert dazu, Sprung für Sprung die Burg seines Erzfeinds zu stürmen. Vorsicht: dieses Spiel macht süchtig!

Panik im Labyrinth: Etwas für Freunde gepflegter Hektik. Mitten in einem Riesenlabyrinth mit herumzappelnden und umherwischenden Energiefressern erwacht Ihre Spielfigur. Und jetzt gilt es, ein knappes Dützend Schlösser zu finden. Das Spiel besticht durch blitzschnelle Bildwechsel und seidenweiche Joystick-Steuerung.

Bulldozer: Eine pfiffige Mischung aus "Pac-Man" und "Soko-Ban", mit liebevoll gestalteter Character-Grafik.

Froggie: "Tierisches" Jump'n-Run-Spiel mit Horizontal-Scrolling. Für 1 oder 2 Spieler simultan.

Soft-Ball: Ein Flipper mit -zig verschiedenen, zockigen Geräuschen — für alle, denen Arzt oder Ehefrau den Besuch der Kneipe an der Ecke verboten haben.

Dow Jones: Ein grafisch aufgepepptes Börsenspiel im EDV-Milieu. Kaufen Sie Atari, IBM oder Amstrad — machen Sie Millionen oder verlieren Sie Haus und Hof.

Persönlichkeitstest: Optisch gelungene "Selbstanalyse" für den kleinen Psychologen. Nicht ganz ernstgemeint, macht aber Spaß.

Skat: Eines der raren Skatprogramme. Fehlen der zweite und dritte Mann für eine zünftige Partie? Dieses Programm hilft aus. Aufwendige Farbgrafik.

Memotron: Die schönste "Memory"-Version, die es je auf einem 8-Bit-Computer gegeben hat. Sie spielen simultan gegen den Computer oder einen menschlichen Gegner und können das gegnerische Spiel durch Ihre eigenen Spielzüge manipulieren. Gutes Gedächtnis und Reaktion werden so gleichermaßen trainiert.

Q-Bert 2: Neue Version des liebenswerten klassischen Perspektiv-Hüpfspiels. Viel Feind, viel Ehr.

The Player's Dream Vol. 6

Ein Paket besonders anspruchsvoller Spielprogramme, die nie langweilig werden — mal flott und kernig, mal tüftelintensiv. **Best.-Nr. 195, DM 19,-***

Soko-Ban Constructor: Eine Herausforderung für Ihre kleinen grauen Zellen. Finden Sie die richtige Vorgehensweise für jedes Spiellevel. Mit Bildschirm-Editor zum Gestalten und Abspeichern eigener Levels. Fünf Levels werden mitgeliefert. Achtung: absolutes Suchtspiel, der Effekt ist "Lemmings"-ähnlich!

Air Traffic Control: Koordinieren Sie den Luftverkehr zwischen acht europäischen Flughäfen! Verhindern Sie Zusammenstöße, indem Sie den Piloten die richtigen Informationen geben. Aufregende Simulation im Grafikmode 2.

Trouble in Space: Strategie und Tempo in einem faszinierenden Spiel vereint. Hier wird nicht geballert, sondern mit kühlem Kopf eine Bahn für einen Space-Scooter gebaut. Sehr hohe Spielmotivation, detailreiche Grafik. Mit Level-Editor.

Super Rubik's Cube: Der berühmte "Zauberwürfel" auf dem CPC-Bildschirm. Mit Spiegeln für die verdeckten Würfelseiten. Für Freunde verzwickter Kombination.

Alien Attack: Auch das klassische Ballerspiel für die Arbeitspause darf nicht fehlen. Diese "Space Invaders"-Version überzeugt durch ihre liebevoll gestaltete Grafik.

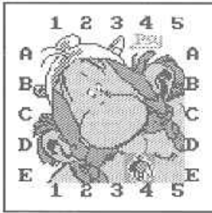
Oik: Temporeiches, "Pacman"-artiges Actionspiel mit niedlich animierter, butterweicher Sprite-Grafik und vielen, vielen Monstern.

Dallas Oil: Management-Gesellschaftsspiel für bis zu 4 Personen. Reizvoll durch ineinander verschachtelte Transaktionen.

Reparator: Flottes Taktik-und Action-Spiel mit farbiger Character-Grafik. Beherrschen Sie das Schlachtfeld!

Gardner-Spiel: Kombination ist angesagt bei diesem Spiel mit Vektoren. Diagonale, senk- oder waagerechte Reihen von Steinen werden auf einem Spielfeld manipuliert. Etwas für Tüftler.

Puzzle — make it!



Neun verschiedene Schiebe-Puzzles zum Entspannen; von den "Ghostbusters" über Conan und Obelix bis hin zum Madonna-Porträt.

255 Schwierigkeits-Stufen. Testen Sie Ihre Vorstellungskraft! **Best.-Nr. 196, DM 15,-***

Codex Vol. 5

Ein Anwenderpaket für kühle Rechner und Freunde des geschriebenen Wortes. **Best.-Nr. 265, DM 19,-***

Textmaker: Ein einfaches Textverarbeitungsprogramm zum Briefeschreiben. Briefkopf gesondert erfaßbar, Blocksatz, Datei-Manager, einstellbarer Tabulator, Zeilenbreite bis 72 Zeichen. Deutsche Schreibmaschinen-Tastaturbelegung mit DIN-Zeichensatz. Basic-Programmierer können die Routinen für deutsche Umlaute und ß auskoppeln und in ihre eigenen Programme einbinden.

Papermaker: Ein Mini-DTP-Programm für die Schüler- oder Vereinszeitung. Mehrspaltiger Text, Schlagzeilen und Zeitungskopf lassen sich frei auf der Seite plazieren. Einfache Geometriegrafik für Zeitungskopf. Köpfe, Schlagzeilen und Artikel getrennt speicherbar. Ausgabe für Epson-kompatible Drucker. Mit einer Demo-Zeitung als Beispiel.

Typograph: Mit diesem Programm entwerfen und verarbeiten Sie Grafikschriften auf dem Bildschirm. Per eingebauter Hardcopyroutine läßt sich die Schriftkunst auch zu Papier bringen. Freies Skalieren und Positionieren der Schriftzeichen. Ein kompletter Font im "Western-Look" wird mitgeliefert — damit Sie gleich loslegen können.

Multitrainer: Ein komfortables Lern-, Übungs- und Test-Programm für beliebige Multiple-Choice-Aufgaben — beispielsweise Führerscheinfragen, Prüfungsaufgaben, Checklisten oder Psycho-Tests. Ergebnisprotokolle speicherbar, Zeitrahmenüberprüfung eingebaut.



Buchomat: Verwaltung aller privaten Finanzen auf kaufmännischer Basis mit allem Komfort. Komplette Übersicht über Aktiva und Passiva. Kontenrahmen mit insgesamt 28 Konten — auch Wertpapiere und Optionen, Steuern, Gebühren, Abschreibungen und Zinsen werden erfaßt. Jahresabschluß, Druckerausgabe für Gesamt- und Vermögensübersicht.

Haushaltsbilanz: Erfassen Sie Ihre laufenden Ausgaben und stellen Sie fest, in welchen Bereichen die Kostenlawine am schlimmsten wütet. Übersichtliche Tabellendarstellung, blitzschneller Überblick über die finanzielle Situation im laufenden Monat. Kinderleichte Menüsteuerung, Druckerausgabe.

Codex Vol. 6

Eine besonders anspruchsvolle Sammlung hilfreicher Anwendungen für verschiedene Lebenslagen. **Best.-Nr. 266, DM 19,-***

Diskettensystem: Die langgesuchte "eierlegende Wollmilchsau". Von den Standard-Datei-Operationen (Formatieren, Kopieren, Directory-Manipulationen) bis hin zum Diskettenoptimierer, File-Reaktivierer und Relokator ist alles eingebaut. Eine Werkbank für die tägliche Arbeit mit Disketten und Dateien. Ein eigenes Benutzermenü läßt sich einklinken — eine Overlay-Datei mit entsprechender Schnittstelle ist dabei.

Platine-Master: Ein sehr aufwendiges Programm zur Gestaltung von Leiterplatten. Joystick-Bedienung mit Icons, Fadenkreuzcursor, Masken für IC-Fassungen auf Abruf, Beschriftung, Kopieren von Bereichen, abschaltbares Hilfsraster. Druckausgabe in zwei Größen.

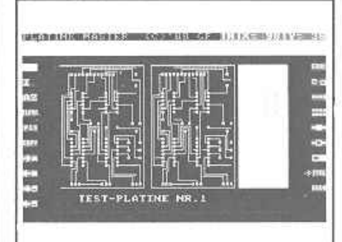
Programmsammlung
Der Datengarten
mit der zusätzlichen Portion
Gebrauchsnutzwert!
3-Zoll-Diskette international CPC
Das Beste von R+E Software
Die Super-Serie
Die DMV-

Terminkalender: Speichert für jeden Tag einen Textbildschirm. Auch für Tagebuch oder Besuchsprotokolle geeignet. Menügesteuert, mit deutschen Umlauten, Druckerausgabe, zahlreichen Editorfunktionen, Hilfebildschirm.

Magic Synthesizer: Hüllkurven und Tonschwebungen erzeugen für ausgefeilte Sounds mit dem Soundchip des CPC. Grafische Darstellung der Hüllkurven, Schritt-für-Schritt-Editieren durch numerische Eingabe möglich. Erzeugt Basic-Code für die Verwendung der Sounds in eigenen Programmen. Demo-Melodie zum Austesten eingebaut.

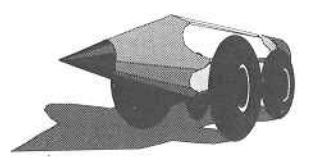
Fußball-Verwaltung: Von der Bundesliga bis zur zweiten Kreisklasse — dieses sehr umfangreiche Programm verwaltet alle Fußballergebnisse komfortabel und erleichtert Tippern die Übersicht. Freie Eingabe und Abspeichern von Ligen und Spielplänen. Das Programm überträgt die einzelnen Paarungen in die Ergebniseingabe. Die Tabellen lassen sich anzeigen und auf dem Drucker ausgeben. Umfangreiche statistische Auswertung.

Lotto-Commander: Erlaubt die Aufstellung und Gewinnkontrolle jeweils eines Service-Scheins der Spiele "6 aus 49" und "Spiel 77" für mehrere Wochen (besonders preisgünstige Tippmethode).



Diskettenhullen-Drucker: Druckt Inhaltsverzeichnis, Archivnummer und Datum auf Hüllen für 3-Zoll-Disketten. Für Epson-kompatible Drucker.

Aufkleber-Editor: Zur Herstellung professioneller, klebefertiger Etiketten für 3-Zoll-Disketten. Sehr aufwendig aufgemacht: "What you see is what you get". Fett-, Kursiv-, Magerschrift, doppelt hohe, breite, unterstrichene und kopfstehende Schrift. Gleichzeitige Verwendung von 3 Zeichensätzen möglich. Mit eingebautem Zeichensatz-Editor. Für Epson-kompatible Drucker.



Die BASIC-Revolution

FAst BAsic COMpiler

Der Turbo-Antrieb für Ihre Basic-Programme!

Haben Sie sich schon immer gewünscht, daß Ihre Basic-Programme auf dem CPC schneller laufen?

Mit dem **FABACOM-Basic-Compiler** wird dieser Wunsch Wirklichkeit: Compilierte Basic-Programme brauchen weniger Platz und laufen schneller ab — weil sie nicht ständig den Basic-Interpreter als "Klotz am Bein" mit dabei haben müssen. Schützen Sie den Code Ihrer Programme vor dem Zugriff Neugieriger, indem Sie Maschinenprogramme daraus machen!

- Voller Sprachumfang von CPC-Basic 1.1 (664/6128)
- Die compilierten Programme sind auf jedem CPC lauffähig
- Separate Compilierung von Programmteilen für Nachladeprogramme
- Fast alle Befehle von CPC-Basic 1.1 auch für CPC 464
- Unterstützung von Integer- und Fließkomma-Arithmetik
- Kompatibel zur Vortex-Peripherie einschließlich RAM-Disk
- Alle CP/M-Dienstprogramme können weiterhin genutzt werden
- Deutsche Bedienungsanleitung und Beispielprogramme

Best.-Nr. 209, DM 49,-*



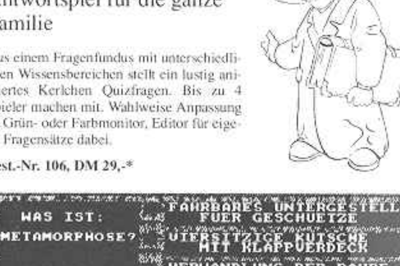
für CPC

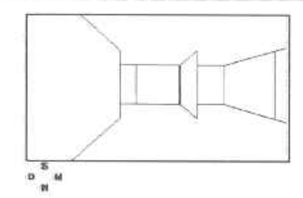
Know CPC

Interessantes Frage- und Antwortspiel für die ganze Familie

Aus einem Fragenfundus mit unterschiedlichen Wissensbereichen stellt ein lustig animiertes Kerlchen Quizfragen. Bis zu 4 Spieler machen mit. Wahlweise Anpassung an Grün- oder Farbmonitor, Editor für eigene Fragensätze dabei.

Best.-Nr. 106, DM 29,-*





Zwei Super-Spiele mit zwei 3D-Brillen für Spielspaß zu zweit

3D-Light Cycle

Das beliebte "Lichtrenner"-Spiel aus dem Film "Tron" in einer holographischen Version für zwei Personen!

3D-Labyrinth

Durch das Anaglyphen-Holographieverfahren wachsen die Labyrinthwände quasi aus dem Bildschirm heraus.

Für alle CPC 464, 664, 6128 mit Farbmonitor. "3D-Light Cycle" kann in 2D-Darstellung auch auf Grünmonitor gespielt werden.

Best.-Nr. 1369, DM 39,-*



Fantastic Four

Vier Superprogramme zum Minipreis

Cockaigne — Weltraum-Action
Terranaut I — Science-Fiction
Fruits — Geschicklichkeit
Terranaut II — Textadventure

Best.-Nr. 1011, DM 29,-*

Special Offers III

Neun Spielprogramme für alle CPCs

Stan und der Zauberstab — Arcade-Adventure
Die Pyramide — Textabenteuer mit Action-Elementen
Puzzle — Zerstreuung für den Feierabend
Black Jack — das klassische Kartenspiel

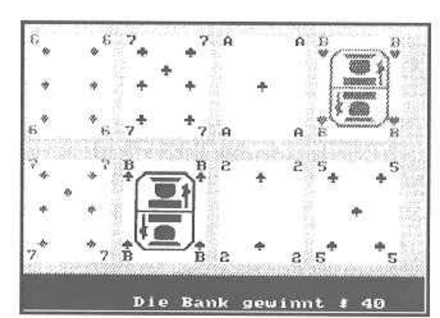
Außerdem: *Orion, Memory, Labyrinth, Zick-Zack, Slalomfahrt*

2 Disketten, Best.-Nr. 107, DM 39,-*

StarTest

Action-Adventure für alle CPCs

Best.-Nr. 104, DM 19,-*



Game Box 3

Für jeden das richtige — vier ausgesuchte Top-Spiele für CPC

- Alphajet: Weltraum-Actionspiel mit tollen Effekten
- *Telefomania:* Science-Fiction mit ganz neuer Spielidee
- *Kampf den Insekten:* erlebnisreiches Geschicklichkeitsspiel
- *Funbouncer:* Farbenfrohes Spiel mit eigenem Bild-Editor

Best.-Nr. 1012, DM 29,-*

Der Schlüssel zum Maschinenraum

Ein CPC-Mega-Sampler, voll mit Werkzeugen für den Assemblerprogrammierer auf dem CPC

Hier finden Sie die besten Tools aus sieben Jahren "PC International" — vom RSX-Manager über den Speicher-Debugger bis zum kompletten Assembler. Ob Sie unter AMSDOS oder CP/M arbeiten — hier ist bestimmt auch für Sie etwas dabei, daß Ihnen Zeit und Mühe sparen hilft.

Aus dem Inhalt:

- Maschinensprachemonitor mit Disassembler
- Ramdisk-Programm für 6128 in Assembler
- Turbo-Relocator: das Organisationsgenie für den Speicher
- "FASTM": Beschleuniger für Mode-1-Bildschirmausgabe
- Maschinensprache-Utilitypaket "Laurel II": Linken, Verschieben, Dokumentieren und RSX-Management
- Utility für Pseudo-Multitasking

Insgesamt über 30 Programme. Best.-Nr. 263, DM 49,-*

CPC-Fraktal-Generator 3D

Vom Maschinensprache-Virtuosen Matthias Uphoff

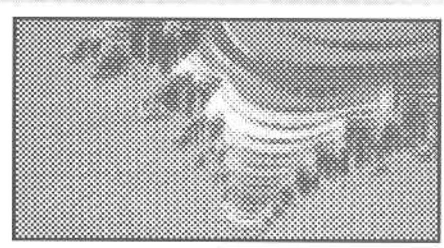
Erleben Sie den Zauber mathematisch erzeugter Landschaften!

- Blitzschnelle Grafik-Erzeugung: Der Fraktal-Generator CPC wurde äußerst effizient in Assembler programmiert. Intelligente Berechnungs-Routinen sorgen für enorme Zeitersparnis. Sie können zuschauen, während sich die fraktalen Seen und Gebirge zügig aufbauen!
- Mandelbrot- und Juliamenge: mit automatischer Glättungsfunktion
- Stufenloser vertikaler Blickwinkel: wahlweise Sicht von oben, unten, schräg und in der Totalen einzeln und stufenlos einstellbar
- Bequeme Handhabung über Pulldown-Menüs mit deutscher Benutzerführung

Vier Bilder können beim 6128 gleichzeitig im Speicher gehalten werden. Die erzeugten Bilder lassen sich mit handelsüblichen Malprogrammen weiterverarbeiten und in eigenen Programmen als grafische Hintergründe einsetzen.

Durch trickreiche Farbrasterung erzeugt der Fraktal-Generator auf dem CPC in Mode 1 zehn verschiedene Farbtöne gleichzeitig.

Best.-Nr. 211, DM 49,-*





Spielend den CPC beherrschen lernen

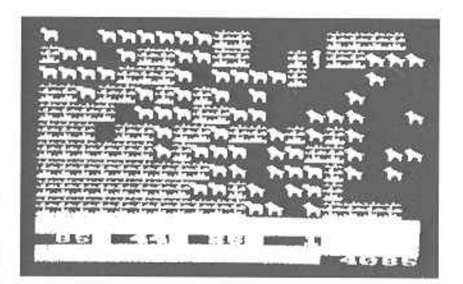
Insgesamt 17 Programme, mit 180seitigem deutschem Handbuch.

Wordmaster: Sie raten ein Wort, das Ihr CPC sich ausgedacht hat. Dann rät der Rechner. Wer braucht weniger Versuche? Sie werden staunen...

Das Spiel des Lebens: Der CPC erzeugt nach den Regeln des natürlichen Wachstums phantastische, kristalline Strukturen. Ändern Sie die Voraussetzungen und sehen Sie, wie neue, nie dagewesene Formen entstehen.

Mit roher Rechengewalt: Mit dem "Brute-Force"-Algorithmus darf der CPC so richtig loslegen und spielt in dem afrikanischen Strategiespiel "Kalaha" alle Gegner an die Wand — oder können Sie ihn schlagen?





Entwicklungshilfe: Eine gut ausgestaltete Simulation versetzt Sie in die Lage eines Entwicklungshelfers, der eine Hungersnot abzuwenden versucht. Aber Ihre Mittel sind begrenzt...

Der CPC als Stratege: Ein raffiniertes System von Spielzugbewertungen macht den CPC zu einem fast unschlagbaren Gegner im Gobang-Spiel, der Ihnen anfangs saftige Niederlagen bescheren wird. Das Programm ist natürlich dennoch zu schlagen — aber wie?

Der Computer lernt: Erleben Sie ein Paradebeispiel für "Künstliche Intelligenz": Das Programm "Minischach" ist lernfähig und wird mit jeder Partie ein bißchen besser.

Das Ökologie-Experiment: Der Computer zeigt Ihnen in animierter Farbgrafik einen simulierten Lebensraum mit Weideland, Schafen und Wölfen. Erforschen Sie die ökologischen Gesetzmäßigkeiten und schaffen Sie ein stabiles, natürliches Gleichgewicht.

Außerdem: *Grafik mit Komplexen Zahlen, Wordketten-Rätsel, Geheimnisse des Pascalschen Dreiecks*

Best.-Nr. 213, DM 39,-*

Programmierte CPC-"Evergreens"

Zwei Disketten mit echten Klassikern fur CPC 464, 664 und 6128! Diese Programme stellen ausgewählte Höhepunkte der letzten Jahre dar, jedes einzelne auf seine Art etwas ganz Besonderes.

CPC-Highlights 1: Spiele-Klassiker

Cave-Flight (das echte "Defender"-Feeling kommt zurück!), *Mad Maze* (Verwirrspiel in drei Dimensionen), *Ortwin's Run* (schnelles Action-Spiel mit Level-Designer), *Panik im Tokio-Express* (spannende Bombensache), *Copter* (Hubschrauber-Geschicklichkeitsspiel), *Bruno Bombe* ("Pacman" läßt grüßen), *Space Taxi* (Spitzen-Jump'n-Run mit Screen-Editor), *Schlange* (Geschicklichkeitsspiel mit Gewöhnungs-Effekt), *3-D-Mühle* (echter Hit! Schlagen Sie den Computer?), *Space Glider* (3-D-Science-Fiction mit Spitzen-Grafik). **Bestell-Nr. 269, DM 49,-**

CPC-Highlights 2: Anwendungen und Utilities

Koma-Copy (fähiges Track-Analyse- und Kopierprogramm), *Easy Printer* (Kreativ-Paket für Grußkarten- und Seitengestaltung), *Digit Extension* (Sprache softwaremäßig digitalisieren mit dem Datenrecorder), *Maskengenerator* (für eigene Abfragebildschirme), *Joystick-Wanze* (Stick-Emulation für Cursortasten), *Minidos* (Disk-Utility), *Interrupt-Musik* (mit RSXen), *Screen-Compressor* (Speicherplatzsparer), *Transfer* (von MS-DOS zum CPC und zurück), *Supermon* (Diskettenmonitor), *Designer* (Zeichensatzgenerator). **Bestell-Nr. 270, DM 49,-**



Futter für PCW-Freunde:



Joyce-Bild-Editor

Zum Zeichnen hochauflösender Grafiken

Der Joyce ist mehr als ein Textverarbeitungssystem! Mit dem Joyce-Bild-Editor gestalten Sie komfortabel hochauflösende Grafiken, binden diese in eigene Programme ein oder geben sie auf dem Joyce-Drucker aus.

Der Joyce-Bild-Editor erschließt die Grafikfähigkeiten Ihres Joyce: Exaktes, pixelgenaues Zeichnen durch Normal- und Feinmodus, Lösch- und Invertierungsfunktion in beiden Bearbeitungsmodi, einfache Integration beliebiger Texte in die Bildelemente, Option zum Mischen von fertigen Bildern, Ausdruckmöglichkeit mit Randbestimmung auf dem Joyce-Drucker, kompletter Quelltext in Turbo Pascal 3.0 wird mitgeliefert.

Joyce-Programmsammlung 4:

Bild-Editor, Best.-Nr. 219, DM 49,-

Joyce-Busineß-Paket

Präsentationsgrafik und Tabellentextverarbeitung

GSX-Plot — die Lösung für Busineß-Grafik auf dem Joyce

Das Programm fertigt Balken- und Kurvendiagramme an; es erlaubt auch die Herstellung von Strich- und Textgrafik. Die Benutzerführung ist menügesteuert. Die fertigen Grafiken lassen sich auf Bildschirm und Drucker ausgeben.

FeldTab — plaziert Ihre Texte millimetergenau

- Komplette Menüsteuerung wie von LocoScript gewohnt
- Wahl von Seitenlänge, Zeilenabstand und Tabulatoren
- Individuelle Bestimmung von Schriftweiten und Schriftarten
- Textübernahme von LocoScript oder im ASCII-Format

Joyce-Programmsammlung 3:

Busineß-Paket, Best.-Nr. 217, DM 59,-*

Joyce-Zeichensatz-Designer und MGX

Zeichensatz-Designer: Editieren, Erzeugen und Abspeichern von Bildschirm-Zeichensätzen — Drucken von vorformatierten ASCII-Texten unter CP/M Plus möglich — vier fertig gestaltete Zeichensätze im Lieferumfang enthalten — Bonusprogramm zum Einstellen der Joyce-Schnittstelle und Tastatur

MGX: Funktionen und Meßwerte grafisch auf Bildschirm und Drucker — arithmetische Grund- und Sonderfunktionen sind darstellbar — gleichzeitige Darstellung mehrerer Funktionen und Meßreihen — Druckerausgabe der Ergebnisse im Großformat möglich — neben einer Klammerebene ist eine Memory-Funktion integriert

Joyce-Programmsammlung 1:

Zeichensatz-Designer und MGX, Best.-Nr. 215, DM 49,-*

GCPM — die grafische Benutzeroberfläche für den Joyce

Schnell, bequem und komfortabel

Der ganze Komfort moderner Computerei — einschließlich Mausbedienung, Uhr, Taschenrechner, Schnittstellensteuerung, Passwortabfrage und vielem mehr.

- Steuern Sie alle Funktionen Ihres Joyce wahlweise über Maus oder Tastatur.
- Arbeiten Sie mit der Darstellung aller Dateien in Tabellenform (ähnlich LocoScript) und den Standardfunktionen wie Kopieren, Umbenennen und Löschen.
- Auf bis zu drei Laufwerken stellen Sie Parameter, Suchpfade (ohne SETDEF) und das temporäre Laufwerk beliebig ein.
- Verändern Sie Dateiattribute, und rufen Sie Programme direkt von GCPM aus auf.
- Nutzen Sie die Funktionen zum Verschlüsseln und Komprimieren von ASCII-Dateien sowie die optionale Passwortabfrage.
- Sparen Sie Zeit durch den integrierten Taschenrechner mit Zwischenspeicher.
- Verändern Sie beliebig die Grundeinstellungen der Schnittstelle, des Zeichensatzes, der Tastatur, der Maus, der Diskettenlaufwerke und des Druckers.
- Ein Screensaver stellt den Monitor bei Nichtgebrauch dunkel.

Mit dem GCPM-Starterset als Installationsprogramm und dem 40seitigen, reich bebilderten Handbuch im Handumdrehen bringen Sie den Komfort der "Großen" im Handumdrehen auch auf Ihren Joyce.

Joyce-Programmsammlung 7:

Grafische Benutzeroberfläche, Best.-Nr. 222, DM 69,-*

Joyce-Tabellenkalkulation

Die universelle Tabellenkalkulation für alle PCWs. Ob Monatsbilanzen, Steuerabrechnung oder Kontoführung — mit diesem Programm sind Sie allen Aufgaben gewachsen.

- Besonders schnelle Berechnungsalgorithmen für die Felderwerte, übersichtliche Menüführung und Abfangen von Fehleingaben
- Am Rand eingeblendete Formel mit bis zu 68 möglichen Zeichen, eigene Befehle zur Verkürzung der Formeln und Exponentialausdrücke

Joyce-Programmsammlung 6:

Tabellenkalkulation, Best.-Nr. 221, DM 59,-*

Joyce-Sonderheft 4

Restposten: nur solange Vorrat reicht!

Aus dem Inhalt:

- WordStar-Verbesserungen
- Bundesligasimulation
- Pulldown-Menüs
- Diskettenmonitor
- Hauptstädte raten in LOGO
- Hardcopy-Routine für 24-Nadler
- dBase-Literaturverwaltung
- Universelles Werkzeug zum Verändern von dBase-Dateien

Heft + DATABOX (2 Disketten), Best.-Nr. 3052, DM 29,-*

Joyce-Datenbank

Die ideale Datenbank für den Joyce mit minimalem Disketten-Speicherbedarf. Dennoch bietet Ihnen die Joyce-Datenbank maximal 27 Felder pro Datensatz, 50 Stellen pro Feld und bis zu 35.000 Datensätze pro Datei.

Daten effektiv verwalten mit der Joyce-Datenbank:

- Besonders schnell durch Verwendung der RAM-Disk für das Programm
- Universell durch die Verwendung von Standard-Direktzugriffsdateien
- Bequem durch Datentransfer von vorhandenen Dateien in neu angelegte
- Maske zum Erfassen und ndern von Daten wird automatisch generiert
- Komfortabel durch Wahl der Sortierkriterien oder Mehrfachsortierung
- Listen mit automatischer Spaltenformatierung und Spaltensummen
- Automatischer Eintrag neuer Dateien ins Disketten-Hauptmenü
- Speicherung einmal gewählter Druckparameter zur Wiederverwendung

Joyce-Programmsammlung 5:

Datenbank für PCW 8256 und 8512 (für PCW 9512 auf Anfrage),

Best.-Nr. 220, DM 69,-*

AMS-Line

Infos von den Werkstatt-Spezialisten

Die Monitore, die zu den CPCs 464, 664 und 6128 gehören, sind robuste Geräte. Dennoch kann es den einen oder anderen guten Grund geben, eine dieser "Flimmerkisten" mal zu öffnen. Vor dem Griff zum Schraubenzieher sollten Sie jedoch diese Folge der AMS-Line lesen – es lohnt sich.

Für die "klassischen CPCs" gibt es Farb- und Grünmonitore in jeweils zwei fast identischen Ausführungen. Der Farbmonitor CTM 644 und sein grüner Kollege GT 65 sind für die CPC-Modelle 664 und 6128 vorgesehen, die über ein eingebautes 3-Zoll-Diskettenlaufwerk verfügen. Dieses benötigt eine eigene 12-Volt-Stromversorgung zusätzlich zu den 5 Volt, mit denen der Rechner gespeist werden will. Zu diesem Zweck haben CTM 644 und GT 65 eine 12-Volt-Anschlußbuchse an der Frontseite. Diese Buchse unterscheidet sie von ihren Vorgängern, die für den CPC 464 gebaut wurden.

Wer einen CPC 664 oder 6128 an einen der älteren Monitore CTM 640 oder GT 64 anschließen will, kann dort die Versorgung des Laufwerks mit wenig Aufwand nachrüsten. Die größte Schwierigkeit könnte darin bestehen, eine geeignete Kupplung oder Buchse für den 2,1-Millimeter-Hohlstecker zu finden, der sich an dem von der Rechnerkonsole kommenden Kabel befindet. Die gängigen Buchsen sind nämlich zu eng. Am besten kauft man das entsprechende Einzelteil bei einem Händler, dem der CPC bekannt ist. Um die benötigte Spannung von 12 Volt abzuzweigen, baut man die abgebildete kleine Schaltung auf. Der Innenleiter der neu erworbenen Kupplung oder Buchse wird mit Masse verbunden, der Außenleiter mit dem +12V-Ausgang der Schaltung. Dann verbindet man die Masse der Schaltung mit dem Minuspol (äußerer Kontakt) der 5-Volt-Versorgung für die Computerkonsole. Zum Schluß wird die Verbindung zwischen Pin 1 des verwendeten ICs und dem "Kontaktpunkt" im Monitor vorgenommen – beim GT 64 ist das der Kollektor von Q 502, beim CTM 640 der Pluspol von C 420.

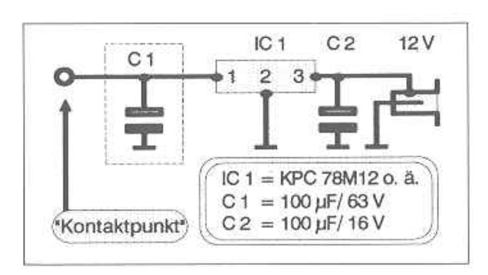
Hin und wieder werden auch die CPC-Monitore Schauplatz technischer Wehwehchen. Oft läßt sich das Problem mit einem simplen Vielfachmeßgerät orten, und auch ein "Nicht-Profi" kann den "Patienten" mit einfachen Mitteln kurieren.

Die folgenden Tips setzen voraus, daß der Monitor zumindest noch ein sichtbares Bild bietet. **Vorsicht!** Aufgrund der in den Geräten erzeugten hohen Spannungen ist eine Berührung aller stromführenden Teile unbedingt zu vermeiden – es besteht Lebensgefahr. Bitte beachten Sie, daß wir keinerlei Verantwortung für irgendwelche eventuell beim Basteln entstehenden Schäden übernehmen können.

Alle Arbeiten mit Ausnahme der Meß- und Einstellvorgänge werden bei gezogenem Netzstecker durchgeführt. Für Spannungsmessungen sind isolierte Klemmspitzen zu verwenden, die angebracht werden, wenn das Gerät stromlos ist. Zum Bewegen der Einstellregler kommen nur nichtmetallische Abgleichwerkzeuge in Frage – beispielsweise eine dünne Plastikstricknadel, die an einem Ende in Schraubenzieherform gefeilt worden ist.

Kleiner Handgriff – große Wirkung

Am CPC 464 wird bei Verwendung eines Diskettenlaufwerks der Controller auf den Expansionsport aufgesteckt. Verschiedene Hardware-Erweiterungen (etwa Speicheraufrüstungen) werden zwischen Rechner und Floppy-Controller installiert. Sollte es beim Einsatz derartiger Erweiterungen zu Disketten-Lesefehlern kommen, könnte eine zu niedrig eingestellte Floppy-Betriebsspannung im Monitor die Ursache sein. Wenn es statt 5V beispielsweise nur 4,8 sind, muß nachjustiert werden. Der entsprechende Regler im Monitor ist VR 501, gemessen wird am Steckkontakt des 5-Volt-Kabels. Keinesfalls darf die Spannung 5 Volt übersteigen, da dies einige ICs übelnehmen würden.



Um eine Floppy-Stromversorgung bei den Monitoren CTM 640 und GT 64 nachzurüsten, baut man die hier dargestellte Schaltung auf und befestigt sie im Inneren des Monitorgehäuses

Sollte die Helligkeit des Monitors nach einiger Zeit zu wünschen übrig lassen und der Helligkeitsregler nicht mehr genug Reserve hergeben, läßt sich mit VR 402 beim CTM und VR 605 beim GT sicher noch etwas herausholen.

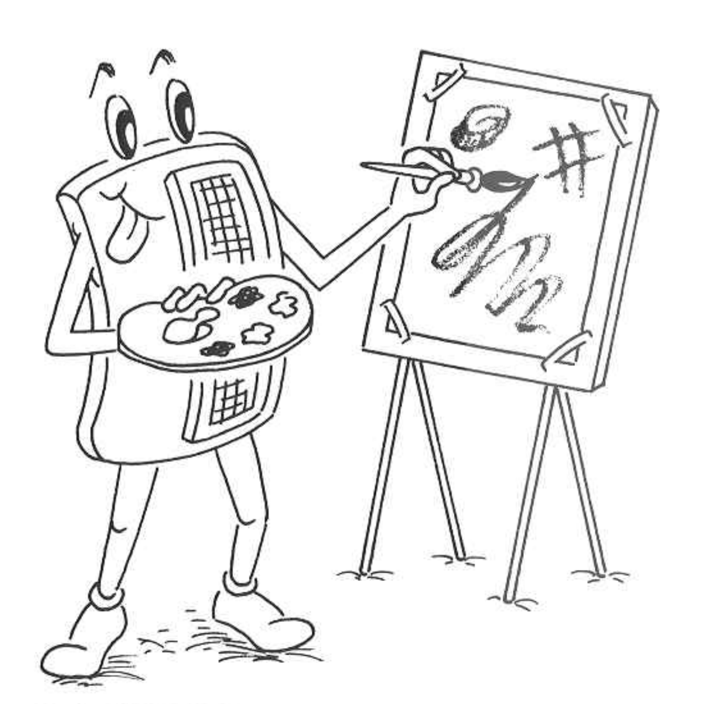
Auch gegen ein unscharfes Bild beim Farbmonitor ist ein Kraut gewachsen: Die Schärfe läßt sich mit dem Fokusregler justieren, der isoliert direkt am Zeilentrafo sitzt. Durch eine Bohrung auf der Rückseite des Monitorgehäuses hindurch ist er mit einem kleinen Schraubendreher zu erreichen, ohne daß man das Gerät öffnen muß. Die Bohrung befindet sich (von der Gehäuserückseite gesehen) rechts unten, etwa 5 cm von der rechten Kante entfernt.

Falls weiße Flächen beim CTM einen deutlichen Farbstich zeigen, ist es Zeit für einen Weißabgleich. Die hierfür vorgesehenen Regler befinden sich auf der Videoplatine, die unmittelbar am Bildröhrensockel angebracht ist. Hier die nötigen Schritte:

– Mit dem Basic-Befehl "BORDER 25" einen weißen Rahmen erzeugen.
– Den Helligkeitsregler nach rechts bis zum Anschlag aufdrehen.
– Die Regler VR 801 und VR 803 bis zum Anschlag nach rechts drehen.
– Die Regler VR 802, VR 804 und VR 805 bis zum Anschlag links drehen.
– Auf der Hauptplatine den kleinen Schiebeschalter suchen und betätigen. Er schaltet die vertikale Ablenkung aus.
– Die Regler VR 802, VR 804 und VR 805 so weit aufdrehen, bis ein waagerechter Strich zu sehen ist. Diesen dann durch wechselseitiges Justieren der drei Regler auf klares Weiß einstellen.
– Schiebeschalter zurückstellen.
– Die Regler VR 801 und VR 803 wechselweise so weit nach links drehen, bis auch der Bildrahmen klar weiß erscheint.

In der nächsten AMS-Line wird es um häufige Defekte bei Monitoren, um Fehlersuche und -beseitigung gehen. Bis dahin wünschen wir Ihnen und Ihrer CPC-Hardware Gesundheit und Wohlergehen.

Rainer Dombrowski/sz



Mit RSX ganz fix

Der Plus bekennt Farbe

Viele der stolzen Plus-Besitzer haben sich den Computer unter anderem wegen der neuen Features, besonders auch wegen der 4096 Farben angeschafft. Enttäuscht mußte man jedoch feststellen, daß nicht alle Fähigkeiten direkt unter Locomotive Basic ausgenutzt werden können. Dahin war der Traum vom Colour Personal Computer, der der CPC normalerweise sein sollte. Aber es gibt noch Lichtpunkte. Unser Farbpack für das Amsdos der Plus-Modelle schafft Abhilfe.

In der CPC 4/5'92 stellte die CPC International Informationen für Assembler-Programmierer bereit, wie der ASIC-Chip und seine Funktionen genutzt werden können. Für BASIC-Programmierer war dies jedoch relativ nutzlos. Zwar konnte nach Lektüre des Artikels das Sprite-Pack genutzt werden, die 4096 Farben blieben jedoch auf der Strecke. Nachfolgende RSX-Erweiterung stellt jetzt mehrere Zusatzbefehle bereit, die allesamt zur Farbdarstellung unter BASIC verwendet werden.

Will man den ASIC unter BASIC programmieren, stehen jede Menge Hürden im Weg, die zuerst genommen werden müssen. So sendet man zuerst die benötigten Codes an den ASIC und poked dann die entsprechenden Farbwerte in den Bereich &6400-&641F. Die Farben sind nun eingestellt, jedoch wird diese Einstellung sofort wieder durch den System-Interrupt zur Farbdarstellung zurückgesetzt. Nachfolgende Überlegung basiert nun darauf, den System-Interrupt durch POKE &6000,&FF zu sperren, wodurch jedoch die Tastaturabfrage gesperrt wird. Und hier sind wir auch schon am Ende der BASIC-Künste angelangt. Was nützen uns die schönsten Farben in eigenen Programmen, wenn die Tastatur nicht mehr genutzt werden kann? Jetzt ist also der Assembler-Programmierer gefragt. Der Trick der abgedruckten Routine besteht nun darin, daß der System-Interrupt praktisch ausgesperrt wird. Man ruft sie also 300mal in der Sekunde über den Fast-Ticker-Interrupt auf. Dadurch wird dem Betriebssystem zuvorgekommen, da der Videochip eine gewisse Zeit benötigt, um die vom Betriebssystem eingetragenen Farben darzustellen.

Die Megacolour-Routine wird aber nun 300mal pro Sekunde aufgerufen. Das führt dazu, daß die Einträge vom Betriebssystem sofort wieder durch unsere ersetzt werden.

4096 Farben

|ASIC.INT

Dieser Befehl veranlaßt das Programm, eine Codefolge zum ASIC-Chip zu senden, um diesen zu aktivieren. Das ist unbedingt nötig, da der CPC plus im Normalzustand nicht in der Lage ist, 4096 Farben darzustellen.

|MC.ON,<Fensterflag>

Das ist der Kernbefehl des ganzen Programms. Er dient dazu, die Farbverwaltung in den Fast-Ticker-Interrupt einzuhängen. Falls der Befehl |ASCI.INT vorher nicht aufgerufen wurde, wird dies hier automatisch nachgeholt. Beim ersten Aufruf, nachdem der M-CODE eingeladen worden ist, werden alle Pens mit augenfreundlichen Grautönen belegt. Bei allen CPCs ist es möglich, alle Pens zurückzusetzen. Dies geschieht normalerweise durch CALL &BC02. Um diesen Call auch im neuen Farbpack nutzen zu können, wurde dieser Vektor auf eine eigene Routine umgebogen.

Um mit dem ASIC-Chip zu arbeiten, wird ein Code gesendet, der den Chip veranlaßt, ein Verwaltungsfenster im Bereich von 4000H bis 7FFFH zu öffnen. In diesem Fenster werden alle neuen Features verwaltet, also auch die Farb- und Sprite-Daten. Falls man nun neben der Farbverwaltung, für die dieses Programm zuständig ist, auch die Sprites benutzen möchte, ist das nicht ohne weiteres möglich, da bei jedem Interruptaufruf das Fenster geöffnet und normalerweise sofort wieder geschlossen wird. Um das Fenster geöffnet zu halten, ist der Parameter Fensterflag da. Wird ein Parameter ungleich null angegeben, bleibt das Fenster geöffnet. Jeder andere Parameter bewirkt, daß das Fenster geschlossen wird.

|MC.OFF

Wird dieser Befehl aufgerufen, wird die Routine aus dem Interrupt ausgeklinkt und der Vektor &BC02 wieder zurückgebogen. Die Farbverwaltung wird dem Betriebssystem wieder übergeben.

|MC.INK,<Farbnummer>,<Rot-Anteil>,<Grün-Anteil>,<Blau-Anteil>

Äquivalent zu dem BASIC-Befehl INK arbeitet auch dieser RSX. Der Bereich für Farbnummer liegt zwischen 0 und 16. Die Nummern von 0 bis 15 gelten als INK-Farben, die genau wie beim (BASIC-)INK-Befehl numeriert sind. Nummer 16 ist der Border-Farbe zugeordnet.

Die Intensität der Farben wird jeweils mit 0 bis 16 angegeben.

Falls bei Verwendung dieses Befehls |ASIC.INT oder |MC.ON noch nicht aktiviert wurden, ist die Einstellung erst nach Aufruf von |MC.ON sichtbar.

|MC.BORDER,<Rot-Anteil>, <Grün-Anteil>, <Blau-Anteil>

|MC.BORDER,<Rot-Anteil>, <Grün-Anteil>,<Blau-Anteil>
Dieser Befehl spricht den Bildschirmrand (Border) an. Die Farbparameter sind genau wie bei |MC.INK anzugeben.

|MC.COPYTAB
Um sich eine Standardpalette zu entwerfen, die bei einem CALL &BC02 eingelesen wird, kann dieser Befehl verwendet werden.

|MC.NORMAL
CALL &BC02 setzt die aktuelle Farbtabelle zurück, |MC.NORMAL die zwischengespeicherten Farbdaten.

Marc Stiegemeyer/rs

```
100 ' MEGACOL.BAS                               [1131]
110 ' CPC 464(plus)/6128(plus)                  [1320]
120 ' (c) 1992 Marc Stiegemeyer                 [1937]
130 '          & CPC International              [2118]
140 ON ERROR GOTO 190:PRINT                     [1891]
150 IF PEEK(&8A00)=&C9 THEN PRINT CHR$(7);      [7204]
"Megacolour V1.0 wurde schon initialisiert
!":GOTO 180
160 MEMORY &8A00-1:LOAD"megacol.bin",&8A00     [3845]
:CALL &8A00:|ASIC.INIT
170 MODE 1:PRINT:PRINT"Megacolour V1.0 ini     [5310]
tialisiert":PRINT
180 PRINT:NEW                                   [833]
190 IF ERR=28 AND ERL=160 THEN PRINT"Megac     [4763]
olour-Befehle greifen nicht!!!"
200 STOP                                        [464]
```

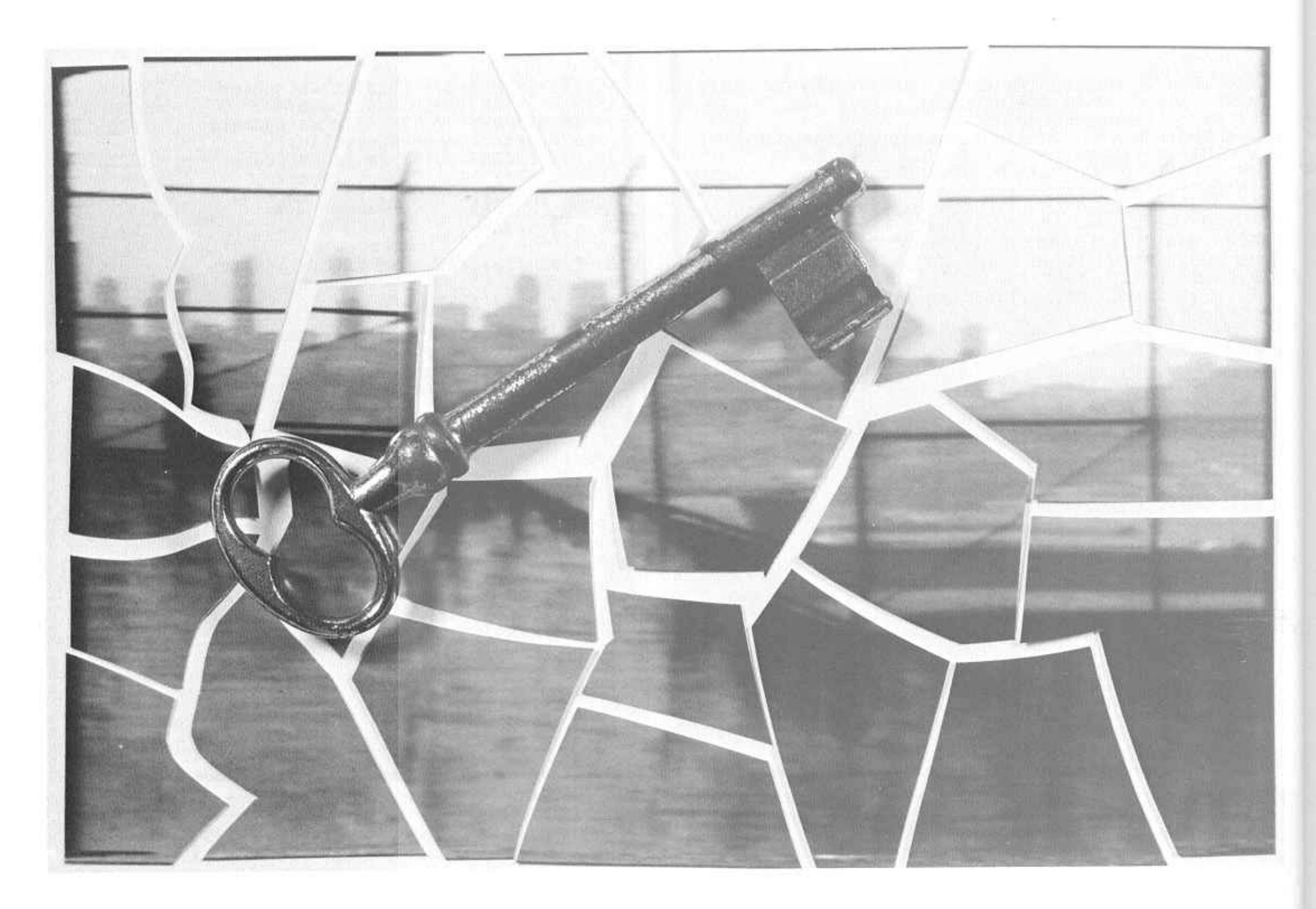
```
100 ' MEGACOL.LAD                               [1218]
110 ' CPC 464(plus)/6128(plus)                  [1320]
120 ' (c) 1992 Marc Stiegemeyer                 [1937]
130 '          & CPC International              [2118]
140 MODE 2:PRINT" Megacolour V1.0":PRINT:P     [4900]
RINT"Bitte warten, checke Datas..."
150 WINDOW 1,80,5,25                            [1034]
160 MEMORY &89FF                                [730]
170 FOR adr=&8A00 TO &8C0F STEP 16              [817]
180   PRINT"Zeile";(adr-&8A00)/16*10+370;"     [2928]
: ";
190   summe=0                                   [552]
200   FOR n=0 TO 15                             [577]
210     READ a$                                 [309]
220     POKE adr+n,VAL("&"+a$)                  [652]
230     PRINT".";                               [665]
240     summe=summe+VAL("&"+a$)                 [1149]
250   NEXT                                      [350]
260   READ summe$                               [567]
270   IF summe<>VAL("&"+summe$) THEN PRINT     [3249]
" ERROR":er=er+1 ELSE PRINT" Okay":ok=ok+1
280 NEXT                                        [350]
290 PRINT:PRINT er;"Fehler und";ok;"richti     [4047]
ge Zeilen."
300 PRINT:PRINT                                 [743]
310 IF (er<>0) AND (ok<>33) THEN PRINT"Ini     [6617]
tialisierung bzw. Abspeicherung nicht moeg
lich...":END
320 PRINT"Abspeichern oder Initialisieren      [5219]
(A/I)? ";:CURSOR 1
330 a$=UPPER$(INKEY$):IF a$<>"A" AND a$<>"     [2339]
I" THEN 330
340 PRINT a$:CURSOR 0:PRINT:PRINT:PRINT"Me     [9369]
gacolour V1.0 ";:IF a$="I" THEN CALL &8A00
:MODE 2:PRINT"initialisiert.":PRINT:NEW
350 SAVE"MEGACOL .BIN",b,&8A00,&209:PRINT"     [5866]
abgespeichert.":PRINT:NEW
360 '                                           [117]
370 DATA F3,3E,C9,32,00,8A,01,0F,8A,21,AA,     [4195]
8B,C3,D1,BC,73,0769
380 DATA 8B,C3,26,8A,C3,43,8A,C3,83,8A,C3,     [3075]
E9,8A,C3,4A,8B,092C
390 DATA C3,CD,8A,C3,DB,8A,F3,3A,C2,8B,B7,     [2929]
C2,65,8B,1E,11,0954
400 DATA 21,B1,8B,06,BC,4E,ED,49,23,1D,20,     [3749]
F9,3E,FF,32,C2,072D
410 DATA 8B,FB,C9,F3,FE,01,20,04,7B,32,C4,     [2251]
8B,3A,C2,8B,B7,089F
420 DATA CC,26,8A,3A,C3,8B,B7,C2,65,8B,3E,     [2789]
FF,32,C3,8B,21,084B
430 DATA 02,BC,11,AE,8B,01,03,00,ED,B0,3E,     [2454]
C3,32,02,BC,21,05BB
440 DATA CD,8A,22,03,BC,21,6A,8B,06,80,11,     [2148]
A7,8A,CD,19,BD,06B9
450 DATA C3,E0,BC,F3,3A,C2,8B,B7,CA,65,8B,     [3411]
3A,C3,8B,B7,CA,0A53
460 DATA 65,8B,21,AE,8B,11,02,BC,01,03,00,     [3033]
ED,B0,AF,32,C3,065E
470 DATA 8B,21,6A,8B,C3,E6,BC,F3,01,B8,7F,     [2194]
ED,49,3E,FF,32,08D6
480 DATA 00,68,21,C5,8B,11,00,64,01,22,00,     [2999]
ED,B0,AF,32,00,04EF
490 DATA 68,FB,3A,C4,8B,B7,C0,01,A0,7F,ED,     [1825]
49,C9,F3,21,E7,097D
500 DATA 8B,11,C5,8B,01,22,00,ED,B0,FB,C9,     [3295]
F3,21,C5,8B,11,07E5
510 DATA E7,8B,01,22,00,ED,B0,FB,C9,F3,FE,     [2663]
02,28,45,FE,04,0858
520 DATA C2,65,8B,DD,5E,00,DD,56,02,DD,4E,     [4108]
04,DD,46,06,78,06F2
530 DATA FE,11,D2,65,8B,79,FE,10,D2,65,8B,     [2675]
7A,FE,10,D2,65,08D9
540 DATA 8B,7B,FE,10,D2,65,8B,21,C5,8B,78,     [2870]
80,47,C5,48,06,0799
550 DATA 00,09,C1,CB,21,CB,21,CB,21,CB,21,     [2996]
79,83,4F,71,23,0659
560 DATA 72,FB,C9,DD,46,02,4A,53,CB,3A,CB,     [3329]
3A,CB,3A,CB,3A,080C
570 DATA CB,BB,CB,B3,CB,AB,CB,A3,18,B5,F3,     [2903]
FE,01,28,12,FE,09DF
580 DATA 03,C2,65,8B,06,10,DD,5E,00,DD,56,     [2463]
02,DD,4E,04,18,0582
590 DATA 9E,06,10,18,D1,3E,07,C3,5A,BB,00,     [2876]
00,00,00,00,00,03BA
600 DATA 00,00,00,41,53,49,43,2E,49,4E,49,     [2208]
D4,4D,43,2E,4F,040F
610 DATA CE,4D,43,2E,4F,46,C6,4D,43,2E,49,     [3624]
4E,CB,4D,43,2E,05C5
620 DATA 42,4F,52,44,45,D2,4D,43,2E,4E,4F,     [3491]
52,4D,41,CC,4D,0592
630 DATA 43,2E,43,4F,50,59,54,41,C2,00,00,     [3573]
00,00,00,00,00,0303
640 DATA 00,FF,00,FF,77,B3,51,A8,D4,62,39,     [2668]
9C,46,2B,15,8A,073C
650 DATA CD,EE,00,00,00,CC,0C,00,00,55,05,     [2346]
88,08,11,01,22,03B1
660 DATA 02,33,03,44,04,66,06,77,07,99,09,     [2388]
AA,0A,BB,0B,DD,0463
670 DATA 0D,EE,0E,FF,0F,AA,0A,06,00,F0,0F,     [3862]
0F,0F,F0,00,FF,05DD
680 DATA 0F,00,00,0F,00,FF,00,06,06,60,06,     [2101]
6F,06,F6,06,00,0300
690 DATA 0F,66,0F,F6,0F,0F,06,06,00,00,00,     [2747]
00,00,00,00,00,01A4
```

```
100 ' MEGACOL.DEM                               [1121]
110 ' CPC 464(plus)/6128(plus)                  [1320]
120 ' (c) 1992 Marc Stiegemeyer                 [1937]
130 '          & CPC International              [2118]
140 ' Initialisierung                           [1300]
150 '                                           [117]
160 KEY DEF 66,0,0,0,0:POKE &BDEE,&C9           [1674]
170 RANDOMIZE TIME                              [1777]
180 DIM ce%(3,3),ca%(3,3)                       [980]
190 ON ERROR GOTO 1070                          [1501]
200 |MC.ON                                      [733]
210 |MC.BORDER,0:FOR n%=0 TO 3:|MC.INK,n%,     [3013]
0:NEXT
220 '                                           [117]
230 ' Aufbau des Demo-Screens Nummer 1          [1970]
240 '                                           [117]
250 MODE 1:LOCATE 11,1:PEN 2:PRINT"M E G A     [5993]
 C O L O U R":LOCATE 8,3:PEN 1:PRINT"D E M
 O N S T R A T I O N"
260 LOCATE 1,6:PEN 2:PRINT"MEGACOLOUR";:PE     [8151]
N 1:PRINT" ist eine Maschinenroutine zur"
270 LOCATE 3,8:PRINT"Darstellung von 16 au     [5145]
s ";:PEN 2:PRINT"4096";:PEN 1:PRINT" Farbe
n."
280 LOCATE 1,10:PEN 3:PRINT"Achtung!!! Das     [4124]
 Programm ist nur auf der"
290 LOCATE 1,12:PEN 2:PRINT"Amstrad plus-S     [4411]
erie (464 plus/6128 plus)"
300 LOCATE 14,14:PEN 3:PRINT"lauffaehig!!!     [2922]
"
310 LOCATE 1,16:PEN 1:PRINT"Die Demo beste     [4339]
ht aus insgesamt 3 Teilen:"
320 LOCATE 14,18:PEN 3:PRINT"1) ";:PEN 2:P     [2999]
RINT"Info-Screen"
330 LOCATE 14,19:PEN 3:PRINT"2) ";:PEN 2:P     [4155]
RINT"Fraktal"
```

```
340 LOCATE 14,20:PEN 3:PRINT"3) ";:PEN 2:P [2922]
RINT"MC-Script"
350 LOCATE 5,23:PEN 1:PRINT"Um zum jeweils [11777]
 naechsten Teil zu":LOCATE 5,25:PRINT"gela
ngen, ist ";:PEN 3:PRINT"[TAB]";:PEN 1:PRI
NT" zu druecken."
360 '                                      [117]
370 ' Farbrandom                           [514]
380 '                                      [117]
390 graphiknr=1                            [795]
400 FOR n1%=1 TO 3:FOR n2%=1 TO 3:ce%(n1%, [6715]
n2%)=ROUND(RND*15):NEXT:NEXT:GOSUB 870:GOS
UB 980:GOTO 400
410 '                                      [117]
420 ' Aufbau des Demo-Screens Nummer 2     [1977]
430 '                                      [117]
440 GOSUB 980:|MC.BORDER,0:FOR n%=0 TO 3:| [4768]
MC.INK,n%,0:NEXT:graphiknr=2
450 CLS:LOAD"fr.pic",&C000:PEN 0:LOCATE 8, [6953]
8:PRINT CHR$(22)CHR$(1)"Weiter mit Druck a
uf [TAB]"CHR$(22)CHR$(0)
460 '                                      [117]
470 ' Farbfelder belegen und darstellen    [2053]
480 '                                      [117]
490 ce%(1,2)=10:ce%(2,2)=5:ce%(3,2)=15:GOS [3573]
UB 870:GOSUB 980 'Gruen
500 ce%(1,3)=10:ce%(2,3)=5:ce%(3,3)=15:GOS [1708]
UB 870:GOSUB 980 'Blau
510 ce%(1,1)=10:ce%(2,1)=5:ce%(3,1)=15:GOS [2740]
UB 870:GOSUB 980 'Rot
520 ce%(1,1)=10:ce%(1,2)=10:ce%(1,3)=10:ce [7527]
%(2,1)=5:ce%(2,2)=5:ce%(2,3)=5:ce%(3,1)=15
:ce%(3,2)=15:ce%(3,3)=15:GOSUB 870:GOSUB 9
80 'Grau
530 ce%(1,1)=10:ce%(2,3)=6:ce%(3,1)=6:ce%( [4645]
3,2)=6:ce%(3,3)=6:GOSUB 870:GOSUB 980 'nor
mal
540 ce%(1,1)=10:ce%(1,2)=10:ce%(2,1)=5:ce% [4331]
(2,2)=5:ce%(3,1)=15:ce%(3,2)=15:GOSUB 870:
GOSUB 980 'Gelb
550 ce%(1,1)=10:ce%(1,3)=10:ce%(2,1)=5:ce% [4834]
(2,3)=5:ce%(3,1)=15:ce%(3,3)=15:GOSUB 870:
GOSUB 980 'Lila
560 GOTO 490                               [371]
570 '                                      [117]
580 ' Aufbau des Demo-Screens Nummer 3     [1976]
590 '                                      [117]
600 graphiknr=3:CLS                        [1243]
610 |MC.BORDER,0:FOR n%=0 TO 3:|MC.INK,n%, [3013]
0:NEXT
620 x%=5:y%=240:FOR n%=3 TO 1 STEP -1:GRAP [3819]
HICS PEN n%:GOSUB 720:x%=x%+4:y%=y%+8:NEXT
630 x%=0:y%=40:FOR n%=1 TO 3:GRAPHICS PEN  [6145]
n%:GOSUB 770:x%=x%+4:y%=y%+8:NEXT
640 x%=73:y%=216:FOR n%=1 TO 3:MOVE x%,y%, [7974]
n%,3:TAG:PRINT "Demonstration - ";CHR$(164
);" 3'1992 by MST";:x%=x%+2:y%=y%+2:NEXT:T
AGOFF
650 '                                      [117]
660 ' Farb-Random                          [578]
670 '                                      [117]
680 FOR n%=0 TO 3:FOR n2%=1 TO 3:ca%(n%,n2 [5502]
%)=ce%(n%,n2%):ce%(n%,n2%)=ROUND(RND*15):N
EXT:NEXT:GOSUB 870:GOTO 680
690 '                                      [117]
700 ' sub: Schriftzuege zeichnen           [634]
710 '                                      [117]
720 MOVE x%,y%:DRAWR 0,120:DRAWR 73,-60:DR [3880]
AWR 73,60:DRAWR 0,-120:MOVER 12,0
730 DRAWR 0,120:DRAWR 146,0:MOVER -146,-60 [5250]
:DRAWR 106,0:MOVER -106,-60:DRAWR 146,0:MO
VER 12,0
740 MOVER 146,100:DRAWR -20,20:DRAWR -106, [7088]
0:DRAWR -20,-20:DRAWR 0,-80:DRAWR 20,-20:D
RAWR 106,0:DRAWR 20,20:DRAWR 0,40:DRAWR -2
0,0:MOVER 32,-60
750 DRAWR 73,120:DRAWR 73,-120:MOVER -37,6 [2494]
0:DRAWR -73,0
760 RETURN                                 [555]
770 MOVE x%,y%:MOVER 95,100:DRAWR -14,20:D [5471]
RAWR -67,0:DRAWR -14,-20:DRAWR 0,-80:DRAWR
 14,-20:DRAWR 67,0:DRAWR 14,20:MOVER 12,-2
0
780 MOVER 95,100:DRAWR -14,20:DRAWR -67,0: [6225]
DRAWR -14,-20:DRAWR 0,-80:DRAWR 14,-20:DRA
WR 67,0:DRAWR 14,20:DRAWR 0,80:MOVER 12,-1
00
790 MOVER 95,0:DRAWR -95,0:DRAWR 0,120:MOV [4118]
ER 107,-120
800 MOVER 95,100:DRAWR -14,20:DRAWR -67,0: [6225]
DRAWR -14,-20:DRAWR 0,-80:DRAWR 14,-20:DRA
WR 67,0:DRAWR 14,20:DRAWR 0,80:MOVER 12,-1
00
810 MOVER 0,120:DRAWR 0,-100:DRAWR 14,-20: [4178]
DRAWR 67,0:DRAWR 14,20:DRAWR 0,100:MOVER 1
2,-120
820 DRAWR 0,120:DRAWR 81,0:DRAWR 14,-20:DR [3773]
AWR 0,-20:DRAWR -14,-20:DRAWR -81,0:DRAWR
81,-60
830 RETURN                                 [555]
840 '                                      [117]
850 ' sub: Farbfeld einblenden             [968]
860 '                                      [117]
870 FOR n%=0 TO 15                         [595]
880 FRAME:|MC.INK,1,ca%(1,1)+n%*((ce%(1,1) [5561]
-ca%(1,1))/15),ca%(1,2)+n%*((ce%(1,2)-ca%(
1,2))/15),ca%(1,3)+n%*((ce%(1,3)-ca%(1,3))
/15)
890 FRAME:|MC.INK,2,ca%(2,1)+n%*((ce%(2,1) [7458]
-ca%(2,1))/15),ca%(2,2)+n%*((ce%(2,2)-ca%(
2,2))/15),ca%(2,3)+n%*((ce%(2,3)-ca%(2,3))
/15)
900 FRAME:|MC.INK,3,ca%(3,1)+n%*((ce%(3,1) [5382]
-ca%(3,1))/15),ca%(3,2)+n%*((ce%(3,2)-ca%(
3,2))/15),ca%(3,3)+n%*((ce%(3,3)-ca%(3,3))
/15)
910 IF brflg<>-1 THEN IF INKEY$=CHR$(9) TH [3006]
EN 1020
920 CLEAR INPUT                            [184]
930 NEXT                                   [350]
940 RETURN                                 [555]
950 '                                      [117]
960 ' sub: Farbfeld aktualisieren          [1018]
970 '                                      [117]
980 FOR n%=0 TO 3:FOR n2%=1 TO 3:ca%(n%,n2 [4640]
%)=ce%(n%,n2%):ce%(n%,n2%)=0:NEXT:NEXT:RET
URN
990 '                                      [117]
1000 ' sub: bei Druck auf TAB Farbfeld akt [6552]
ualisieren, und naechster Screen
1010 '                                     [117]
1020 FOR n2%=1 TO 3:FOR n3%=1 TO 3:ce%(n2% [7659]
,n3%)=ca%(n2%,n3%)+n%*((ce%(n2%,n3%)-ca%(n
2%,n3%))/15):NEXT:NEXT:GOSUB 980:brflg=-1:
GOSUB 870:brflg=0
1030 IF graphiknr=1 THEN 440 ELSE IF graph [7472]
iknr=2 THEN 600 ELSE |MC.OFF:MODE 2:CALL &
BC02:PEN 1:POKE &BDEE,&C3:KEY DEF 66,0,252
,252,252:END
1040 '                                     [117]
1050 ' sub: Error-Handling                 [600]
1060 '                                     [117]
1070 IF ERR=32 AND ERL=450 THEN CALL &BC02 [9320]
:MODE 2:PRINT"Das Fraktalbild ist auf Disk
ette nicht zu finden!!!":GOTO 1100
1080 IF ERR=28 AND ERL=200 THEN IF PEEK(&8 [16917]
A00)<>&C9 THEN CALL &BC02:MODE 2:PRINT"Meg
acolour V1.0 wurde noch nicht initialisier
t!!!":GOTO 1100 ELSE PRINT"Megacolour V1.0
-Befehle greifen nicht!!!":GOTO 1100
1090 RESUME NEXT                           [703]
1100 POKE &BDEE,&C3:KEY DEF 66,0,252,252,2 [2819]
52:|MC.OFF:STOP
```

```
100 ' MEGACOL.MIX                          [1202]
110 ' CPC 464(plus)/6128(plus)             [1320]
120 ' (c) 1992 Marc Stiegmeyer             [1713]
130 '          & CPC International         [2118]
140 '                                      [117]
150 ' Initialisierung                      [1300]
160 '                                      [117]
170 ON BREAK GOSUB 750                     [768]
180 DIM inten(3):nummer=1                  [840]
190 '                                      [117]
200 ' Bildschirm einrichten und aufbauen   [2013]
210 '                                      [117]
220 |MC.OFF:MODE 0:WINDOW 1,20,1,8:FOR n%= [4169]
0 TO 15:INK n%,0:BORDER 0:NEXT
230 PEN 9:PRINT" MegaColour - "+CHR$(164)+ [3510]
" MST"
240 FOR n%=2 TO 4                          [583]
250 GRAPHICS PEN n%:DRAWR 0,10:DRAWR 180,0 [4834]
:DRAWR 0,-10:DRAWR -180,0:MOVER 4,4:FILL n
%: ' Auswahlbalken
260 MOVER -4,20,n%+9:DRAWR 0,100:DRAWR 180 [8708]
,0:DRAWR 0,-100:DRAWR -180,0:MOVER 4,4:FIL
L n%+3:MOVER 224,-28: ' R/G/B-Sichtfenster
270 NEXT                                   [350]
```

```
280 MOVE 0,152,1:DRAWR 638,0:DRAWR 0,100:D [7261]
RAWR -638,0:DRAWR 0,-100:MOVER 4,4:FILL 8:
 ' Misch-Sichtfenster
290 TAG:FOR n%=0 TO 2:MOVE 62+n%*228,140,9 [3289]
:PRINT"00";:NEXT:TAGOFF: ' R/G/B-Anteile
300 PEN#1,10:LOCATE#1,5,9:PRINT#1,"Gesamt: [3563]
 0000": ' Mischfarbe
310 |MC.BORDER,0:|MC.INK,0,0:|MC.INK,1,136 [4617]
5:|MC.INK,2,3840:|MC.INK,3,240
320 |MC.INK,4,15:|MC.INK,5,0:|MC.INK,6,0:| [5750]
MC.INK,7,0:|MC.INK,8,0:|MC.INK,9,2730
330 |MC.INK,10,1638:|MC.INK,11,1365:|MC.IN [5239]
K,12,1365:|MC.INK,13,1365:|MC.INK,14,1092
340 |MC.ON                                  [733]
350 '                                       [117]
360 ' Erste Abfrage                         [742]
370 '                                       [117]
380 LOCATE 3,4:PEN 14:PRINT"Farbwahl:  R G [2486]
 B"
390 a$=UPPER$(INKEY$)                       [789]
400 IF INSTR("RGB",a$)=0 OR a$="" THEN 390 [2084]
410 '                                       [117]
420 ' Zweite Abfrage                       [1188]
430 '                                       [117]
440 PEN 14:LOCATE 2,4:PRINT"Auswahl: "+CHR [4427]
$(242)+CHR$(32)+CHR$(243)+" R G B"
450 LOCATE 4,6:PRINT"RETURN SPACE I"       [1367]
460 |MC.INK,nummer+10,1365:nummer=INSTR("R [6458]
GB",a$):|MC.INK,nummer+10,4095
470 GOSUB 590                               [966]
480 a$=UPPER$(INKEY$):IF a$="" THEN 480    [1820]
490 IF a$=CHR$(242) THEN IF inten(nummer)> [5243]
0 THEN inten(nummer)=inten(nummer)-1: ' <-
- (Farbkomponente Minus 1)
500 IF a$=CHR$(243) THEN IF inten(nummer)< [4379]
15 THEN inten(nummer)=inten(nummer)+1: ' -
-> (Farbkomponente Plus 1)
510 IF a$=CHR$(13) THEN inten(nummer)=0: ' [4169]
 RETURN/ENTER (Farbkomponente:=0)
520 IF a$=CHR$(32) THEN inten(nummer)=15:  [2645]
' SPACE (Farbkomponente:=15)
530 IF a$=CHR$(73) THEN inten(nummer)=15-i [3758]
nten(nummer): ' I (Komp. invert.)
540 IF INSTR("RGB",a$)<>0 THEN 460: ' R/G/ [2767]
B-Komponenten-Anwahl
550 GOTO 470                                [454]
560 '                                       [117]
570 ' sub: Balken- und Farbdarstellung      [679]
580 '                                       [117]
590 FOR n%=0 TO 1                           [570]
600 FOR n2%=0 TO 1                          [507]
610 MOVE (nummer-1)*228,n%*8+n2%*2,1:DRAWR [2640]
 inten(nummer)*12,0
620 IF inten(nummer)<>15 THEN DRAWR 180-in [2991]
ten(nummer)*12,0,(nummer+1)
630 NEXT                                    [350]
640 NEXT                                    [350]
650 |MC.INK,nummer+4,inten(nummer)*16^(3-n [3153]
ummer)
660 |MC.INK,8,inten(1),inten(2),inten(3)   [3577]
670 TAG:MOVE 62+(nummer-1)*228,140,9:IF in [8280]
ten(nummer)<10 THEN PRINT"0";MID$(STR$(int
en(nummer)),2); ELSE PRINT MID$(STR$(inten
(nummer)),2);
680 TAGOFF                                 [1066]
690 farbe=inten(1)*256+inten(2)*16+inten(3 [1976]
)
700 LOCATE#1,5,9:PRINT#1,"Gesamt: ";RIGHT$ [4445]
("0000"+MID$(STR$(farbe),2),4);
710 RETURN                                  [555]
720 '                                       [117]
730 ' sub: ESCape                           [529]
740 '                                       [117]
750 MODE 1:|MC.OFF:CALL &BC02:PEN 1:PAPER  [3453]
0:END
```



Der Schlüssel zum Code

Alkatraz 2 – Flucht aus der Codierung

Alkatraz wußte schon in der Ausgabe 12/1'92 für Aufregung zu sorgen. Zum einen, weil dieser Kopierschutz dort genau beschrieben wurde, zum anderen –, ja es hat auch etwas mit der Schreiberei zu tun – weil es etwas mit der Rechtschreibung gehapert hat. Aber jetzt ganz richtig: Alkatraz mit z und nicht mit x.

Geht man davon aus, daß jeder Kopierschutz, der von schlauen Gehirnen erdacht wurde, kein wirklicher Schutz vor Hackern und Softwarepiraten ist, ist dies doch sehr erschreckend. Trotzdem, eine Menge Wissen ist nötig, um bestimmte Schutzmechanismen auszuschalten. Der Standard-Computeranwender hat also keine Chance, sich in das Denkmuster der Coder sowie im Gegenzug der Hacker einzuschalten, da hier fundierte Kenntnisse der Maschinensprache erforderlich sind.
Um auch Ihnen einen Einblick zu verschaffen, haben wir in der Ausgabe 12/1'92 den Alkatraz-Kopierschutz vorgestellt und gleich eine kleine Routine zum Verschlüsseln eigener Daten mitgeliefert. Was jedoch tun, wenn die Daten verschlüsselt sind, eine Entschlüsselung jedoch erwünscht ist? Ganz einfach: Den Alkatraz-Deprotector abtippen und starten. Alles andere geht fast wie von selbst.

(K)eine Chance für den Hacker?

Sind beide Programme eingegeben und unter den Namen DECODE.BAS sowie DECODE.LAD abgespeichert, sollte zuerst Listing 2 gestartet werden. Es erzeugt jetzt eine Binärdatei mit dem Namen DECODE.BIN, die von dem Hauptprogramm benötigt wird.
Startet man nun das Hauptprogramm, fragt es nach dem Namen der zu entschlüsselnden Datei. Erfolgt hier keine Angabe, wird davon ausgegangen, daß sich das zu entschlüsselnde Programm schon im Speicher befindet. Nun geben Sie nur noch die Startadresse des Binärcodes sowie den Namen der nach der Decodierung abzuspeichernden Datei ein. Bei zweiter Eingabe gilt wieder, daß keine Abspeicherung erfolgt, sondern die Decodierroutine einfach verlassen wird. Per Hand können nun die entschlüsselten Daten auf Kassette oder Diskette gesichert werden.

Alkatraz entschlüsselt

Die Decodierung an sich geht nach folgendem Schema vor:
Zuerst erfolgt ein Test, der feststellen soll, ob das Programm überhaupt mit einem Alkatraz kompatiblen Codiermechanismus verschlüsselt ist. Ist dieser gefunden, wird die erste Schleife identifiziert, und Decode ermittelt den Codiertyp, mit welchem die Routine verschlüsselt wurde. Alkatraz liegt ja

– wie bekannt – in vier unterschiedlichen Variationen mit jeweils voneinander abweichenden Codierungen vor. Ist die erste Schleife decodiert, wird gegebenenfalls eine weitere freigelegt, die dann auch typengerecht behandelt wird. Dieser Vorgang ist beendet, sobald keiner der vier Schleifentypen erkannt werden kann.

Leicht entschlüsseln

In der CPC 12/1'92 wurde lediglich auf den ersten Schleifentyp eingegangen. Da es jedoch noch drei weitere gibt, hier eine kleine Erläuterung:
Während der erste Schleifentyp mit einem festen Wert arbeitet, tun dies die drei anderen nicht. Als Schlüsselwert benutzt man hier kein festes im Programmcode enthaltenes Byte. Man nimmt vielmehr das Refresh-Register des Z80 zu Hilfe. Es erhöht sich nach jedem Taktzyklus des Z80 um den Wert 1. Lädt man also zunächst einen festen Wert in dieses Register und liest es nach einer genau definierten Zeit wieder aus, erhält man einen veränderten Wert, der ausschließlich von der Laufzeit des Programms abhängig ist. Jede noch so geringe Veränderung des Programms hat zwangsläufig Einfluß auf die Anzahl der benötigten Taktzyklen. Nutzt man nun den so erhaltenen Wert zur Decodierung, läßt das falsch entschlüsselte Programm den Computer über kurz oder lang abstürzen. Ein sicherer Ablauf kann also nicht garantiert werden.
Unsere Decodierschleife ermittelt jedoch den Schlüsselwert genau, und so werden alle vier Routinen richtig entschlüsselt.

William Wieprecht/rs

```
100 ' DECODE.BAS                             [695]
110 ' CPC 464(plus)/664/6128(plus)          [1940]
120 ' (c) 1992 William Wieprecht            [1093]
130 '          & CPC International          [2118]
140 ON ERROR GOTO 590:CLEAR                 [1364]
150 MODE 1:INK 1,26:INK 0,0:BORDER 0:INK 2 [6661]
,13:INK 3,6:PEN 1:PAPER 0:CLS:LOAD"DECODE.
BIN",&AF00:zyk=6:lp=0
160 LOCATE 7,2:PRINT"Alkatraz Deprotection [14737]
 System":LOCATE 10,4:PEN 2:PRINT"written b
y  ";:PEN 3:PRINT"RED DRAGON":PEN 1:LOCATE
 5,10:PRINT"Schleife Nr. :":LOCATE 5,12:PR
INT"Codierrotine :"
170 LOCATE 1,20:INPUT"Programmname : ",na1 [3685]
$
180 INPUT"Startadresse : ",adr:IF adr<0 TH [3954]
EN adr=adr+65536
190 IF adr<&1000 THEN PRINT CHR$(7)," Unzu [4738]
laessig !":LOCATE 1,21:GOTO 180
200 MEMORY adr-1:IF na1$="" THEN 210 ELSE  [1624]
LOAD na1$,adr
210 IF PEEK(adr)=&F3 AND PEEK(adr+6)=&4F T [4769]
HEN space=adr+7 ELSE LOCATE 18,24:PRINT"ER
ROR!":RUN'TEST
220 'Identifizierungsschleife               [1206]
230 lp=lp+1:typ=0:m=space+10                [1623]
240 IF PEEK(m)=&57 AND PEEK(m+1)=&77 AND P [2494]
EEK(m+2)=&23 THEN typ=1
250 IF PEEK(m)=&1B AND PEEK(m+1)=&7A AND P [4013]
EEK(m+2)=&B3 THEN typ=2
260 IF PEEK(m)=&E1 AND PEEK(m+1)=&ED AND P [2943]
EEK(m+2)=&5F THEN typ=3
270 IF PEEK(m)=&ED AND PEEK(m+1)=&5F AND P [4911]
EEK(m+2)=&FD THEN typ=4
280 IF typ=0 THEN WINDOW 1,40,7,25:CLS:WIN [3089]
DOW 1,40,1,25:LOCATE 1,12:GOTO 320
290 LOCATE 20,10:PRINT lp:LOCATE 20,12:PRI [3920]
NT typ:ON typ GOSUB 370,420,470,530
300 GOTO 230                                 [423]
310 'Speichern + Ende                        [760]
320 PRINT"Startadresse : &";HEX$(space),,, [10333]
,,:INPUT"Programmname :",na2$:IF na2$="" T
HEN GOTO 340 ELSE SAVE na2$,B,space,lx-1
330 IF na2$="" THEN GOTO 340 ELSE SAVE na2 [1420]
$,B,space,lx-1
340 LOCATE 3,25:PEN 2:PRINT"I wish you a n [7786]
ice byte !  ";:PEN 3:PRINT"RED DRAGON":CAL
L &BB06:CALL &BB4E:CALL &BBFF:END
350 'Decodierung                             [919]
360 'TYP 1                                    [82]
370 wert=PEEK(space+7):POKE &AF37,wert:lo= [4334]
PEEK(space+1):hi=PEEK(space+2)
380 POKE &AF31,lo:POKE &AF32,hi:lo=PEEK(sp [3746]
ace+4):hi=PEEK(space+5)
390 lx=lo+hi*256:POKE &AF34,lo:POKE &AF35, [4338]
hi:CALL &AF30
400 space=space+18:zyk=zyk+(lx)*9:zyk=zyk+ [3321]
3:GOSUB 580:RETURN
410 'TYP 2                                    [89]
420 zyk=zyk+2:GOSUB 580:POKE &AF01,zyk      [2282]
430 lo=PEEK(space+1):hi=PEEK(space+2):POKE [4344]
 &AF06,lo:POKE &AF07,hi
440 lo=PEEK(space+4):hi=PEEK(space+5):lx=l [7165]
o+hi*256:POKE &AF09,lo:POKE &AF0A,hi:CALL
&AF00
450 space=space+16:zyk=zyk+(lx)*9:GOSUB 58 [2281]
0:RETURN
460 'TYP 3                                    [88]
470 zyk=zyk+5:GOSUB 580:POKE &AF46,zyk      [2031]
480 lo=PEEK(space+7):hi=PEEK(space+8):POKE [5642]
 &AFF1,lo:POKE &AFF2,hi
490 lo=PEEK(space+1):hi=PEEK(space+2):POKE [4757]
 &AF4B,lo:POKE &AF4C,hi
500 lo=PEEK(space+4):hi=PEEK(space+5):lx=l [5971]
o+hi*256:POKE &AF4E,lo:POKE &AF4F,hi:CALL
&AF45
510 space=space+25:zyk=zyk+(lx)*13:GOSUB 5 [1855]
80:RETURN
520 'TYP 4                                    [95]
530 zyk=zyk+4:GOSUB 580:POKE &AF7D,zyk      [2239]
540 lo=PEEK(space+5):hi=PEEK(space+6):POKE [6389]
 &AF89,lo:POKE &AF8A,hi
550 lo=PEEK(space+1):hi=PEEK(space+2):lx=l [6123]
o+hi*256:POKE &AF85,lo:POKE &AF86,hi
560 lo=PEEK(space+8):hi=PEEK(space+9):POKE [4813]
 &AF82,lo:POKE &AF83,hi:CALL &AF7C
570 space=space+28:zyk=zyk+(lx)*15:GOSUB 5 [2575]
80:RETURN
580 za=zyk:za=INT(za/128):zyk=zyk-za*128:R [3626]
ETURN
590 LOCATE 1,24:PRINT CHR$(7);"FEHLER "ERR [5470]
" in Zeile "ERL:CALL &BB06:LOCATE 1,24:PRI
NT SPACE$(39):RESUME NEXT
```

```
100 ' DECODE.LAD                             [694]
110 ' CPC 464(plus)/664/6128(plus)          [1940]
120 ' (c) 1992 William Wieprecht            [1093]
130 '          & CPC International          [2118]
140 FOR a=&AF00 TO &AFB4:READ a$:POKE a,VA [8505]
L("&"+a$):b=b+VAL("&"+a$):NEXT:IF b<>19037
 THEN PRINT"Fehler beim abtippen der DATAz
eilen !" ELSE SAVE"DECODE.BIN",b,&AF00,&B5
150 END                                      [110]
160 DATA 3E,08,32,2C,AF,21,F0,3F,01,10,02, [1561]
C5,7E,57,3A,2C
170 DATA AF,5F,06,09,3C,FE,80,C2,1C,AF,D6, [3101]
80,10,F6,32,2C
180 DATA AF,7B,AA,77,23,C1,0B,78,B1,20,E0, [2336]
C9,18,01,00,00
190 DATA 21,00,00,01,00,00,16,00,7E,AA,57, [3047]
77,23,0B,78,B1
200 DATA 20,F6,C9,00,00,3E,09,32,F0,AF,21, [2752]
30,40,01,10,00
210 DATA 3A,F0,AF,C5,57,06,0D,3C,FE,80,C2, [2488]
5F,AF,D6,80,10
220 DATA F6,32,F0,AF,7E,AA,57,3A,F1,AF,AA, [3479]
57,3A,F2,AF,AA
230 DATA 77,00,00,23,C1,0B,78,B1,C2,50,AF, [3480]
C9,3E,0A,32,2D
240 DATA AF,21,45,87,01,02,02,FD,21,FF,3F, [2829]
C5,3A,2D,AF,FD
250 DATA AE,00,AC,AD,FD,77,00,0B,2B,FD,23, [2261]
3A,2D,AF,06,0F
260 DATA 3C,FE,80,C2,A8,AF,D6,80,10,F6,32, [2767]
2D,AF,C1,0B,78
270 DATA B1,C2,8B,AF,C9                     [1280]
```

Gamer's Message

Schummeln gilt doch!

Allmählich müßte es sich in Spielerkreisen herumgesprochen haben: Für Tips und Tricks rund um alte und neue CPC-Spiele und Spielchen ist die Gamer's Message da. Deshalb keine lange Vorrede, los geht's!

Der Schrecken aller Mailboxbetreiber sind sicher Hacker. Wer nicht auf echte Datenreisen aus ist und lieber **HACKER II** spielt, erhält hier die Codes: 00987, Cover, Yashar, Wogan und Hippy. Noch nie war hacken so einfach ...

Kurztips für Oldies ...

Auch für **CHASE H.Q.** haben wir etwas auf Lager. Wählen Sie die Funktion Tasten definieren, und geben Sie die Buchstaben S,H,O,C,K,E und D ein. Während des Spiels haben nun die Tasten 1 bis 4 eine besondere Bedeutung. 1 ist ziemlich uninteressant – Neubeginn des Levels. Mit 2 gelangt man in den nächsten Teil, 3 zeigt die Endgrafik an, und 4 versorgt Sie mit einem neuen Leben. Wer kennt ihn noch, den süßen Roboter Nummer 5, der auch Star von **SHORT CIRCUIT** war? Aus dem ersten Teil helfen Sie ihm, indem Sie gleichzeitig die Tasten des Wortes OCEAN drücken.

Spindizzy

Einen speziellen Gag gibt es bei **SPINDIZZY**. Wer hier während des Titelbildes gleichzeitig F und X drückt, wird mit einer Grafik erfreut. Diese läßt sich auch noch mit einigen Tasten beeinflussen. Schon ein paar Jährchen hat **GHOSTHUNTERS** auf dem Buckel –war es doch eines der ersten Spiele des Erfolgssoftwarehauses Codemasters. Weitgehend unbekannt ist aber, daß man in einen Cheat-Modus gelangt, wenn man DEL und die Tasten COMPLEX drückt. Mit Hilfe der Buchstabentasten lassen sich nun die verschiedenen Räume wählen.
Und für alle Fans von Komplettlösungen: **VENDETTA**. Gleich im ersten Bild befindet sich in einer der Kisten eine Drahtschere. Nun die Tür aufkicken und den Raum betreten. In einer Kiste befinden sich Granaten. Eine Kiste weiter rechts kann man in den Besitz eines Maschinengewehrs kommen. In einer Kiste in der Ecke befindet sich Munition. Jetzt kann man sich endlich mit dem Maschinengewehr bewaffnen und den Raum wieder verlassen. Ein Raum weiter rechts befindet sich eine Leiter.

Vendetta

Diese hochsteigen und das Gebäude betreten. Wer sucht, wird hier ein Buch finden. Weil dieser Raum mit Medien aller Art vollgestopft zu sein scheint, halten Sie auch nach einer Diskette und einer roten Videokassette Ausschau. Im nächsten Raum befinden sich ein Buch mit einem Code und ein blaues Videoband. Jetzt gehen Sie an den Computer, legen die Diskette ins Laufwerk und geben den aus dem Buch entnommenen Code ein. Nun erhalten Sie wieder ein Buch. Im nächsten Raum sollten Sie sich der Kiste annehmen, die links vom Radio steht. Einen Raum weiter hat jemand seine Autoschlüssel vergessen. Sie liegen auf dem Tisch, vor dem zwei Stühle stehen. An der Munitionsschachtel läßt sich wieder das nötige Blei für das Maschinengewehr auftreiben. Einen Raum weiter, und Sie finden ein Bett. Nicht einschlafen, sondern das Kissen in Augenschein nehmen. Jetzt kann man das Gebäude verlassen, hinuntersteigen und sich aus einer Kiste eine kugelsichere Weste besorgen. Nun die Autoschlüssel anwählen, möglichst nahe an den rechten Reifen des Autos herangehen, nach links oben schauen und die Tür öffnen (das geht wie das Aufnehmen von Gegenständen).

Der Rächer des Bruders

Jetzt kommt erst mal eine kleine Autofahrt, bevor es im dritten Teil weitergeht.
Die Tür, die man nun findet, öffnen und dann durchschreiten. Nun sucht man nach einer Handtasche, einer Karte und einem Buch. Dann kann man das Zimmer verlassen. Zwei Räume weiter rechts findet man neue Munition. Rechts geht's ins Innere des Gebäudes, wo sich wieder mal ein Kissen befindet. Auch dieses wird untersucht, anschließend macht man sich wieder aus dem Staub. Man wählt den Schlüssel an und öffnet die Tür nach dem gleichen Prinzip wie beim Auto. Nun drehen Sie eine Runde in Ihrem Flitzer, bis Sie am Flughafen ankommen. Wie gewohnt zuerst nach innen gehen. Dort ergattern Sie eine Verkleidung und eine weitere kugelsichere Weste. Jetzt bis ins Cockpit vorstoßen, die Karte mitnehmen und die Drahtschere anwählen. Nun erscheint eine Bombe. Diese kann entschärft werden, indem die Drähte in der richtigen Reihenfolge durchgeschnitten werden. Zuerst der mittlere, dann der linke, zuletzt der rechte.
Jetzt ist es bald geschafft. Im Park bis zum Springbrunnen gehen, und Sie erblicken Ihren Bruder, der auf dem Boden liegt. Niederknien, ihn befreien und ... geschafft!!!

B.A.T. – Hilfe naht

In Heft 6/7'92 veröffentlichten wir einen Hilferuf von Mika Töwe. In UbiSofts Science-Fiction-Adventure **B.A.T.** kam er nicht in Crisa Kortakis Wohnung – der Wächter wies ihn immer ab. Hilfe kommt von Sven Fischer aus Hamburg. "Crisa", schreibt Sven, mag keine unerwarteten Besucher. Wenn man ihn jedoch vorher anruft und sein Kommen ankündigt, dann bereitet auch der Wächter keine Schwierigkeiten. Die Telefonnummer bekommt man übrigens von Lydia". Vielen Dank für den Tip – Mika's Problem müßte damit eigentlich gelöst sein.
Vor dem ersten Kampf in **SAMURAI WARRIOR** empfiehlt es sich, von der Übungsoption Gebrauch zu machen und die Steuerung zu üben. Was nirgends in der Anleitung erwähnt wird: Mit gezogenem Schwert springt unser Kämpfer weiter und höher. Nur so kann man manche Hindernisse überwinden.
Die erfolgversprechendste Taktik bei **URIDIUM**: Zuerst säubert man die Oberfläche der Raumschiffe von statischen Objekten. Wenn dann noch nicht genügend zerstört ist, legt man sich irgendwo auf die Lauer und wartet auf gegnerische Formationen, die eine

leichte Beute sind. Schlagkräftigere Angreifer läßt man einfach vorbeiziehen.

Der Rächer wartet auf seine Opfer

Das größte Problem sind wahrscheinlich die Satelliten, die Ihr Raumschiff verfolgen. Wenn möglich, sollte man sich aus dem Staub machen, bevor sie starten. Ist es dazu zu spät, muß man versuchen, den Verfolger auszubremsen und unerwartet die Richtung zu ändern. Wer panisch abhaut, findet sich meist in der nächsten Mauer wieder.

SAINT DRAGON kann man auf zwei verschiedene Arten spielen. Entweder ganz normal und fair. Dann ist das Spiel ziemlich schwer und verursacht leichte Frustgefühle. Oder man kuschelt sich so in den Schwanz des Drachen ein, daß dieser alle Schüsse abfängt. Dann ist die Sache ein wahres Kinderspiel, und man muß eigentlich nur in der linken oberen Ecke bleiben und den Feuerknopf betätigen.

Saint Dragon

Wenn's bei Action-Spielen an Leben oder an Energie fehlt, helfen oft Tastenkombinationen weiter. Doch nicht alle Programmierer geben dem Käufer gleich Schummelhilfen mit. Dann helfen meist nur Pokes, für die man entweder ein Multiface II benötigt, oder Listings, die den Kopierschutz umgehen und direkt ins Programm poken. Aber nicht jeder hat ein Multiface, und die Schummellistings sind lang, kompliziert und deshalb rar. Aller guten Dinge sind drei, haben wir uns gedacht, und deshalb stellen wir hier und heute eine weitere Möglichkeit vor, für die man nur einen Sectoreditor mit Suchfunktion benötigt.

Sie suchen einfach eine vorgegebene Zahlenfolge auf der Diskette und ersetzen sie durch eine andere. Um bei **Steeve McQueen Westphaser** unendlich viel Energie zu erhalten, ersetzen Sie einfach die Folge 91,30,01,AF durch 00,30,01,AF.

Genauer erklärt sei dies für den Sectoreneditor von Discology, da es sich hier um den komfortabelsten und wahrscheinlich auch gebräuchlichsten handelt. Nach dem Starten wählen Sie den Editor. Dort im Menü MODES wählen Sie EDITION DISQUE und stellen als Starttrack 0, als Endtrack 41 und als aktuellen Track 0 ein (auf französisch heißt Track *Piste*).

Um die Suche zu beginnen, wählen Sie im Menü FONCTIONS den Punkt RECHERCHER, was Suchen bedeutet. Die zu suchende Kombination ist Hexadezimal, deshalb drücken Sie nun H. Jetzt können Sie die Zahlenkette eingeben, nach der gesucht werden soll – in unserem Beispiel also 91 30 01 AF. Nach einiger Zeit zeigt das Programm nun an, wo die Zeichenkette gefunden wurde. Schreiben Sie diese Adresse auf. Nun gehen Sie ins Menü und wählen mit Druck auf C die Funktion COURANT an. Mit den Cursortasten bewegen Sie nun den Cursor an die vorher aufgeschriebene Adresse (die Adresse, an der sich der Cursor momentan befindet, wird rechts unten angezeigt). Sind Sie an der gewünschten Adresse angelangt, ersetzen Sie die alten Werte durch die neuen. In unserem Fall also durch 00 30 01 AF. Dann in der Menüleiste durch Anwählen von ECRIRE speichern.

Diskette unter die Lupe nehmen bringt Lebensenergie

Etwas langsamer, aber einfacher geht es auch mit dem Programm HANDY MAN. Dort ist das Teilprogramm "Disc Search" zuständig, auch hier gibt man die Zahlen hexadezimal ein und läßt das Programm dann suchen. Der Cursor befindet sich nach der Suche automatisch an der richtigen Stelle, so daß die Zahlenfolge nun nur noch überschrieben werden muß. Danach natürlich Speichern nicht vergessen!

Ebenfalls recht verbreitet unter unseren Lesern dürfte das Programm DOC sein, das vor einiger Zeit in der CPC International abgedruckt war. Hier stellt man zuerst die Suche nach Hex-Codes ein (F0 schaltet zwischen ASCII und Hexadezimal um). Nach Drücken des Punktes am Zehnerblock wird die Länge der Zeichenkette eingegeben (zwei Ziffern zählen als eine Zahl), dann die zu suchenden Zeichen. Sobald das Programm die Suche beendet hat, ändern Sie die entsprechende Stelle und speichern mit F2. Sind so alle Veränderungen vorgenommen worden, verlassen Sie den Editor und starten das Spiel wie gewohnt – nur eben mit ungewohnten Erleichterungen. Viel Spaß!

Achtung, diese Methode der Sektormanipulation verändert Ihre Diskette! Deshalb sollten Sie zuerst eine Sicherheitskopie anfertigen und nur mit dieser hantieren. Sonst können Sie das Spiel nie wieder spielen, ohne zu schummeln. Im schlimmsten Fall, wenn Sie einen Fehler gemacht haben, kann auf diese Weise das Programm sogar ganz zerstört werden.

Probleme, die auftauchen könnten: Discology meldet: "Piste YY non formatée". Dann wird einfach der Starttrack bei EDITION DISQUE um eins erhöht. Wenn Discology "Secteur trop grande réduit à &4000" meldet, sollte man das Unternehmen aufgeben, da sonst das Programm zerstört werden könnte.

The Crypt

Für **THE CRYPT** haben wir folgende Kombinationen ausgegraben:
82,F2,82,69 wird durch 00,F2,82,69 ersetzt, danach wird nach 35,21,70,6F gesucht. Statt dessen setzen Sie 00,21,70,6F ein. Zum Schluß verfahren Sie mit 05,F2,11,7C genauso. Die neue Folge lautet hier 00,F2,11,7C.

Im nächsten Heft gibt's Kombinationen dieser Art für eine ganze Reihe weiterer Spiele.

Abschließend noch ein Wort zu den Anfragen, die bei uns eingehen. Wir versuchen, bei Ihren Problemen zu helfen. Aber bitte geben Sie doch genau an, wo Sie festhängen. Wenn wir Briefe bekommen, in denen nur allgemein nach Tips gefragt wird ist es – speziell bei Adventures –nicht gerade einfach, dem Schreiber zu helfen.

Andreas Lober/jg

An dieser Stelle wieder die Aufforderung an unsere Leser: Schreiben Sie uns, wenn Sie bei irgendeinem Spiel auf dem CPC festhängen. Und lassen Sie uns Tips wissen, die Sie beim Spielen herausgefunden haben. Tastenkombinationen, Komplettlösungen, Tips, Fragen und natürlich Antworten — alles ist willkommen.

Unsere Adresse:
DMV-Verlag
Redaktion CPC International
Kennwort: Gamer's Message
Fuldaer Straße 6
W-3440 Eschwege



INPUT einmal anders

Neue Eingaberoutine als RSX

Was machen, wenn der Befehl INPUT zu unkonventionell, eine eigene Eingaberoutine in BASIC zu langsam und viel zu komplex ist? Na, wer weiß es? Natürlich, wir schnappen uns die CPC Amstrad und tippen ein paar Zeilen ab, die uns mehr Komfort und Geschwindigkeit bei der Zeicheneingabe garantieren – Fehleingaben natürlich ausgeschlossen.

Obwohl die Tastatur beim CPC noch als wichtigestes Eingabemedium gilt, funktioniert gerade die Eingabe nicht immer so komfortabel, wie man es sich eigentlich wünschen würde. Doch dies muß nicht an der Tastatur liegen. Schuld ist in den meisten Fällen die Eingaberoutine, welche zum Beispiel keine Rückspünge zum Zeilenanfang oder gar eine Umschaltung vom Einfüge- zum Überschreibmodus erlaubt. Dem Autoren dieses Programms ist es mit genau 671 Bytes gelungen, einen perfekten Ersatz für das doch recht spärliche INPUT-Kommando des Locomotive Basic zu schaffen.
So können jetzt schon bestehende Strings weiterverarbeitet und zulässige Tasten vordefiniert werden. Ebenso erlaubt der String-Editor, die Länge sowie die Priorität für Groß- und Kleinschreibung festzusetzen. Individuell wird schon beim Aufruf zwischen dem Überscheib- und Einfügemodus gewählt, und auch das Zeichen/die Taste zum Verlassen der Eingabeaufforderung läßt sich vordefinieren. Beim Verlassen der Routine wird das zuletzt eingegebene Zeichen in eine beliebige Variable eingetragen. Nach Belieben läßt sich jetzt auch die Positionierung des Eingabefeldes direkt beim Aufruf übergeben. Das spart einen weiteren LOCATE-Befehl und somit wertvollen Speicherplatz.
Zwar mag der Aufruf der Routine mit seinem Parameterwulst etwas ungelenk erscheinen, betrachtet man jedoch die eben genannten Vorteile, läßt sich dieser "Mehraufwand" bei der Programmierung leicht rechtfertigen.

Alles ist erlaubt

Aber schauen wir uns erst einmal die einzelnen Parameter etwas genauer an.
Direkt am Anfang wird durch den Parameter **mode** der Eingabemodus vordefiniert. Eine 0 schaltet direkt in den Überschreibmodus, 1 wählt den Einfügemodus an.
capslock läßt Befehle wie UPPER$ oder LOWER$ überflüssig werden. Die hier zur Verfügung stehenden drei Werte stehen für den Beibehalt der Originalschreibweise (0), Umwandlung aller eingegebenen Werte in Klein- (1) oder in Großschreibung (2).
Sehr wichtig ist auch der nachfolgende Parameter **char%**. Da sich – wie sich

```
ISTRINGEDIT,mode,capslock,@c
har%,@ascii$,@text$,spalte,zeile
```

Korrekte Aufrufsyntax für den String-Editor mit allen Parametern







<RETURN>/<ENTER>	Eingabe abschließen
<→>	Cursor nach rechts
<←>	Cursor nach links
<SHIFT>+<→>	Cursor ans Ende
<SHIFT>+<←>	Cursor an den Anfang
<DEL>	Zeichen vor Cursor löschen
<CLR>	Zeichen unter Cursor löschen
<CONTROL>+<E>	Umschaltung Einfüge-/Überschreibmodus
<CONTROL>+<→>	und
<CONTROL>+<←>	und
<↓>	und
<↑>	und
<SHIFT>+<↓>	und
<SHIFT>+<↑>	und
<CONTORL>+<↓>	und
<CONTORL>+<↑>	und
<ESC>	Abbruch

Innerhalb der Eingaberoutine stehen folgende Tastenfunktionen zur Verfügung

beim nächsten Parameter herausstellen wird – alle erlaubten Zeichen für die Eingabe vordefinieren lassen, muß natürlich auch ein Zeichen bestimmt werden, durch das die Eingabe beendet wird. Im Normalfall wird hier das Zeichen 13 (Taste <ENTER>) oder 27 (Taste <ESC>) stehen. Denkbar ist aber auch die Tastenkombination <CONTROL>-<X>, die sich durch das Zeichen 24 an die Variable **@char%** übergeben läßt. Die genauen Werte für die einzelnen Tastendefinitionen erfahren Sie in Ihrem Benutzerhandbuch (ANHANG III, Zeichnung "ASCII Standardwerte").

Wie schon erwähnt, lassen sich die bei einer Eingabe erlaubten Zeichen vorbestimmen. Legen Sie also alle erlaubten Zeichen in der Variable **ascii$** ab. Sollen zum Beispiel nur Zahlenwerte mit möglicher Punkttrennung eingegeben werden, ist in den String **ascii$** die Zeichenkette "0123456789." zu übergeben.

Als nächstes steht noch die Variable **text$** an. Sie enthält nach erfolgter Eingabe den eingegebenen Text. Wichtig hierbei ist noch, daß eine maximale Stringlänge, wie Sie vor dem Aufruf bestand, zugrunde liegt. Soll also ein String mit einer Länge von maximal 40 Zeichen editiert werden, der vor dem Aufruf leer ist, muß **text$** zuvor mit `text$=STRING$(40," ")` auf die entsprechende Länge gebracht werden.

Für die Positionierung der Eingabeaufforderung stehen nun noch die sinnigerweise mit **spalte** und **zeile** bezeichneten Parameter bereit. Die Wertbereiche orientieren sich hier an dem Bildschirmmodus und sind äquivalent zu den bei LOCATE zu verwendenden Parametern.

Hubert Cronenbrock/rs

```
100 ' String-Editor Datengenerator             [1807]
110 ' CPC 464(plus)/664/6128(plus)             [1940]
120 ' (c) 1992 Hubert Cronenbrock              [1380]
130 '         & CPC International              [2118]
140 MEMORY &9FFF                               [134]
150 FOR adr=&A000 TO &A29E STEP 11             [1861]
160   FOR i=adr TO adr+10                      [979]
170     READ b$                                [315]
180     byte=VAL("&"+b$)                       [465]
190     POKE i,byte                            [294]
200   NEXT i                                   [375]
210 NEXT adr                                   [547]
220 SAVE"str-edit.bin",b,&A000,&29E            [1707]
230 END                                        [110]
240 DATA 08,A0,C3,23,A0,C3,23,A0,53,54,52     [1914]
250 DATA 49,4E,47,45,44,49,D4,53,C5,01,00     [1387]
260 DATA A0,21,9B,A2,CD,D1,BC,3E,C9,32,14     [1125]
270 DATA A0,C9,FE,07,C2,7F,A2,DD,6E,00,CB     [1422]
280 DATA 7D,C2,84,A2,3E,19,BD,DA,84,A2,DD     [1447]
290 DATA 66,02,CB,7C,C2,84,A2,3E,50,BC,DA     [1837]
300 DATA 84,A2,4C,22,97,A2,DD,6E,04,DD,66     [1895]
310 DATA 05,7E,FE,00,CA,7F,A2,32,91,A2,23     [1882]
320 DATA 5E,23,56,ED,53,95,A2,47,81,DA,84     [1952]
330 DATA A2,FE,51,28,03,D2,84,A2,DD,7E,0C     [2319]
340 DATA FE,00,28,02,3E,01,32,93,A2,DD,6E     [1241]
350 DATA 08,DD,66,09,22,99,A2,7E,32,94,A2     [1111]
360 DATA 2A,97,A2,CD,75,BB,EB,7E,CD,5A,BB     [2213]
370 DATA 23,10,F9,3E,01,32,92,A2,2A,97,A2     [1560]
380 DATA CD,75,BB,CD,81,BB,CD,09,BB,38,FB     [1994]
390 DATA CD,06,BB,4F,FE,0D,CA,31,A2,2A,99     [2275]
400 DATA A2,77,FE,F3,CA,80,A1,FE,F2,CA,92     [1616]
410 DATA A1,FE,F7,CA,A1,A1,FE,F6,CA,C4,A1     [1714]
420 DATA FE,7F,CA,D2,A1,FE,10,CA,F9,A1,FE     [1425]
430 DATA 05,CA,17,A2,FE,EF,CA,31,A2,3E,EF     [1114]
440 DATA B9,30,06,3E,FB,B9,D2,31,A2,DD,7E     [2486]
450 DATA 0A,FE,00,28,25,FE,01,28,0F,3E,60     [2450]
460 DATA B9,30,1C,3E,7A,B9,38,17,3E,DF,A1     [1485]
470 DATA 18,0D,3E,5A,B9,38,0D,3E,40,B9,30     [1710]
480 DATA 08,3E,20,B1,4F,2A,99,A2,77,3A,94     [2474]
490 DATA A2,B9,CA,31,A2,DD,6E,06,DD,66,07     [1752]
500 DATA 7E,FE,00,CA,7F,A2,23,5E,23,56,47     [1590]
510 DATA 1A,B9,28,06,13,10,F9,C3,CA,A1,3A     [2135]
520 DATA 93,A2,FE,01,28,1B,3A,92,A2,5F,16     [879]
530 DATA 00,2A,95,A2,19,2B,71,3A,91,A2,BB     [2834]
540 DATA CA,35,A2,1C,7B,32,92,A2,C3,35,A2     [1697]
550 DATA 3A,92,A2,47,3A,91,A2,5F,90,C5,F5     [2492]
560 DATA 4F,06,00,16,00,2A,95,A2,19,2B,E5     [2607]
570 DATA D1,F1,28,0C,2B,ED,B8,C1,EB,04,78     [1833]
580 DATA 32,92,A2,18,01,C1,71,C3,35,A2,3A     [1422]
590 DATA 91,A2,47,3A,92,A2,B8,28,40,3C,32     [1605]
600 DATA 92,A2,3E,09,18,3A,3A,92,A2,FE,01     [822]
610 DATA 28,31,3D,32,92,A2,3E,08,18,2B,3A     [2236]
620 DATA 92,A2,47,3A,91,A2,4F,90,FE,00,CA     [1164]
630 DATA A5,A0,CD,84,BB,47,3E,09,CD,5A,BB     [1749]
640 DATA 10,FB,79,32,92,A2,CD,81,BB,C3,A5     [1900]
650 DATA A0,CD,84,BB,C3,92,A0,3E,07,CD,5A     [1838]
660 DATA BB,C3,A5,A0,3A,92,A2,FE,01,28,F1     [1947]
670 DATA 4F,06,00,2A,95,A2,09,2B,E5,D1,1B     [2183]
680 DATA 3A,91,A2,91,3C,4F,ED,B0,3E,20,12     [1498]
690 DATA 3A,92,A2,3D,32,92,A2,C3,35,A2,3A     [1660]
700 DATA 92,A2,4F,06,00,2A,95,A2,09,E5,D1     [2215]
710 DATA 1B,3A,91,A2,91,4F,FE,00,28,02,ED     [2274]
720 DATA B0,3E,20,12,C3,35,A2,3A,91,A2,FE     [2355]
730 DATA 01,CA,A5,A0,3A,93,A2,FE,01,20,07     [1656]
740 DATA AF,32,93,A2,C3,CA,A1,3E,01,18,F6     [1529]
750 DATA CD,84,BB,C9,CD,84,BB,2A,97,A2,CD     [1709]
760 DATA 75,BB,2A,95,A2,3A,91,A2,47,7E,CD     [1666]
770 DATA 5A,BB,23,10,F9,2A,97,A2,3A,92,A2     [2252]
780 DATA 84,3D,67,C3,9A,A0,50,61,72,61,6D     [1323]
790 DATA 65,74,65,72,2D,46,65,68,6C,65,72     [2053]
800 DATA 20,21,57,65,72,74,62,65,72,65,69     [1710]
810 DATA 63,68,73,2D,46,65,68,6C,65,72,20     [2216]
820 DATA 21,21,58,A2,18,03,21,6A,A2,7E,CD     [2018]
830 DATA 5A,BB,23,FE,21,20,F7,C9,C9,01,01     [1920]
840 DATA 00,00,B6,8B,15,20,EB,21,96,A2,00     [1617]
```

```
100 ' String-Editor Laderoutine                 [1256]
110 ' CPC 464(plus)/664/6128(plus)              [1940]
120 ' (c) 1992 Hubert Cronenbrock               [1380]
130 '         & CPC International               [2118]
140 MEMORY &9FFF                                [134]
150 LOAD"str-edit.bin"                          [1671]
160 CALL &A014                                  [631]
170 PRINT "STRingEDIT installiert.":NEW         [3403]
```

CPC

"Digiglobe" und Druckerprobleme

Das Programm "Digiglobe" von der Databox 2-3/92 läuft auf meinem CPC 464 mit Emulator einwandfrei. Leider gibt mein Epson-Drucker immer erst etliche kurze Zeilenvorschübe aus, bevor der Städtename gedruckt wird. Und noch etwas: Ist das Programm "Copyshop" auch auf Kassette erhältlich?

Wolfgang Karl, Coburg

Wir vermuten, daß Ihr Drucker bei jedem ausdrücklichen Linefeed immer einen doppelten Zeilenvorschub ausgibt. Dieses Problem tritt recht häufig und in Verbindung mit den unterschiedlichsten Programmen auf. Es macht sich in der Regel durch sinnlos eingeschobene Leerzeilen bei Text- oder Grafikausgabe bemerkbar. Was also tun bei überflüssigen Zeilenvorschüben?
Stellen Sie den DIP-Schalter für CR/LF (Linefeed bei jedem Return ausgeben) auf OFF. Falls dies keine Abhilfe schafft, müssen Sie Pin 14 des Druckerkabels ("Autofeed/XT") durchtrennen. Dann sollte kein überflüssiger Zeilenvorschub mehr erfolgen. Wenn wider Erwarten alles nichts fruchtet: Schalten Sie Ihren Drucker in den Hexdump-Modus und machen Sie einen "Digiglobe"-Ausdruck. Ermitteln Sie anhand einer ASCII-Tabelle (Druckerhandbuch!) die Hexwerte des ausgegebenen Städtenamens und machen Sie diese Hexwertereihe im ausgedruckten Hexdump ausfindig. Die dieser Reihe vorausgehenden Hexwerte sind dann die "Übeltäter": Steht dort der Wert "0A" (dezimal 10), so hat Ihr CPC ausdrücklich ein "Linefeed"-Zeichen geliefert. Dieses läßt sich unterdrücken, indem man die entsprechende Ausgabe von CHR$(10) im Listing des entsprechenden Programms ausfindig macht und eliminiert.
Zu Ihrer zweiten Frage: "Copyshop" und andere kommerzielle CPC-Programme des DMV-Verlags sind nicht (mehr) auf Kassette erhältlich. Die Überprüferei und die Erledigung von eventuellen Reklamationen fressen so viel teure Arbeitszeit, daß die Sache sich nicht lohnt. Immerhin treten bei Kassetten um ein Vielfaches mehr Datenfehler auf als bei Diskettensoftware. Darüber hinaus haben die meisten CPCler inzwischen eine Diskettenstation – die Preise für gebrauchte Disk-Laufwerke können heute auch kaum noch jemanden vom Hocker schubsen.

CPC

Neue Hardware für den CPC?

Ihr baut ganz schön ab! Ihr solltet mal eine Erweiterung entwickeln (lassen), die 256 kByte RAM + 8 Epromsockel und RS-232-Schnittstelle und eventuell noch anderes enthält. Das Ganze mit Management als RSX im EPROM. Das wäre nach 8 Jahren CPC schon längst überfällig!

G. Hannemann, Kocherstetten

Schade, wenn Leser der Meinung sind, wir würden abbauen. Es ist kein Geheimnis: Der CPC ist schon seit Jahren zum Nischencomputer geworden. Warum wohl haben alle anderen kommerziellen Zeitschriften in Deutschland, die sich mit ihm beschäftigten, schon vor Jahren dichtgemacht? Wir versuchen, das kleiner gewordene Grüppchen der CPCler und PCWler auch weiterhin zusammenzuhalten und mit interessanten Informationen zu versorgen. Auf diese Weise sind wir inzwischen automatisch selbst von einem großen Standard-Computer-Magazin (1986) zum Nischenblatt (1992) geworden – mit viel, viel bescheideneren Mitteln als früher, einer total zusammengeschrumpften Auflage und einem winzig kleinen Redaktionsteam.
Oh – ja: Nur zu gern würden wir ein Mammut-Hardware-Projekt wie das von Ihnen vorgeschlagene auf die Beine stellen. Wir – das heißt, zwei Redakteure und ein Chefred, müßten uns dafür allerdings ein paarmal kopieren. Das hat blöderweise bis jetzt noch nicht so richtig geklappt. Oder wir müßten einen Spitzen-Hardware-Entwickler auf dem freien Markt einkaufen, ihn dazu überreden, seine PC-Technik zu vergessen und eine Auftragsarbeit von drei Mann-Monaten Umfang für uns zu übernehmen. Kleinigkeit – und da die ganze Redaktion regelmäßig Lotto spielt, kann es auch nicht mehr lange dauern, bis wir uns das leisten können.
Jetzt aber mal im Ernst: Wir bemühen uns, eine gute Zeitschrift zu machen, sind jedoch noch viel stärker als früher auf die Mitarbeit unserer Leser und freien Mitarbeiter angewiesen. Und die machen normalerweise in erster Linie das, was ihnen selber Spaß macht.
Vielleicht findet sich ja ein Hardwarefreak, der so etwas wie die von Ihnen vorgeschlagene Kombi-Aufrüstung schon einmal gebaut hat und uns anbietet. Wenn die Sache sicher nachbaubar ist und wir ein halbwegs lesertaugliches Platinenlayout daraus machen können, sind wir gern dabei.

PCW

KERMIT-Bezugsadressen

Sie beschrieben in Ausgabe 11'89 die Kopplung eines PCWs mit einem PC durch das PD-Programm "Kermit". Können Sie mir sagen, wo ich diese Software herbekomme?

Erwin Sättler, Feldafing

Das Datenübertragungsprogramm "Kermit" ist Public-Domain-Software und darf somit frei verteilt werden. "Kermit" gibt es in Versionen für so ziemlich jeden Computer. Die PCW-Version ist auf der PCW-DATABOX-Diskette 11'89 verfügbar. CPC- und PC-"Kermit" gibt es auf den entsprechenden CPC- beziehungsweise PC-DATABOXEN zur gleichen Ausgabe. Freilich gibt es das Programm auch einzeln bei verschiedenen PD-Anbietern und in Mailboxen. Eine Bezugsquelle, bei der man nicht bloß die üblichen PC-Disketten, sondern auch CP/M-Software für CPC und PCW bekommt: Public-Domain-Initiative, Postfach 11 18, 6464 Linsengericht.

CPC

COMAL für CPC 6128

Ich suche Ansprechpartner zum Thema Comal für CPC 6128. Meine Frau ist Lehrerin und möchte Comal gerne im Unterricht einsetzen, leider fand sich in der Schule aber kein Handbuch mehr dazu auf. Ein Schreiben an die Comal-Anwendergruppe brachte nur "unbekannt verzogen". Auch die Suche in Kleinanzeigenteilen brachte nicht den gewünschten Erfolg. Können Sie mir weiterhelfen?

Heino Hansen, Flensburg

Wie unsere Recherchen ergeben haben, besteht die "Comal User Group" wohl nicht mehr. Allerdings gibt es im Handel nach wie vor Comal-Systeme, wenn auch nur noch für PCs, Amiga und C-64. Die Firma AK modul bus, die sich auf Lernsysteme für Schulen spezialisiert hat, kümmert sich um Comal-Benutzer. Ansprechpartner ist hier Herr Hagemann. Wenn Sie Dokumentation benötigen, können Sie auch zu zwei Büchern greifen, die sicherlich noch hier und da zu haben sind: Fischers "Comal in Beispielen" (1986 erschienen bei Teubner, 208 Seiten, DM 24,80) und Buschs "Comal für Einsteiger" (1987 erschienen bei Franzis, 115 Seiten, DM 38,–).

CPC PLUS

Kompatibilitätsprobleme

Ich trage mich mit dem Gedanken, mir einen CPC Plus zu kaufen. Meine Frage ist nun, ob die alte Software auch wirklich 100% auf den neuen läuft, da schon vermehrt über Probleme berichtet wurde. In Heft 2-3/92 haben Sie ein Kompatibilitätsmodul vorgestellt, das bei Schwierigkeiten helfen soll. Werden damit alle Probleme behoben, und was kostet es?

Peter Krichbaum, Darmstadt

Leider gibt es tatsächlich etliche Programme, die Probleme auf den neuen Modellen machen. So verweigern etwa "Mastercopy", "Discology", "Zap t'Balls Advanced" und ältere Versionen vieler Spiele die Zusammenarbeit mit den Plus-Rechnern. Das gleiche gilt für einige Anwenderprogramme, die Betriebssystemfunktionen des CPC direkt anspringen. Es gibt ja durchaus einige Systemunterschiede zwischen den klassischen 6128ern und dem Plus – so muß etwa die direkte Tastaturabfrage für den letzteren um zwei Befehle erweitert werden.
Das von Ihnen erwähnte Kompatibilitätsmodul bringt laut Herstellerangaben etwa 40% der "Problemsoftware" auf dem Plus zum Laufen. Ein weiteres Kompatibilitätsmodul eines anderen Herstellers ist in Arbeit,

mehr läßt sich darüber jedoch noch nicht sagen. Eine softwaremäßige Anpassung der nicht lauffähigen Programme wäre zwar möglich, aber nur mit einem enormen Zeitaufwand zu verwirklichen: Ganze Programme müßten disassembliert, gepatcht und neu assembliert werden. Daran wird sich wohl niemand machen, denn mit einer solchen Sträflingsarbeit ließe sich auf dem relativ kleinen CPC-Plus-Markt auch kein Geld verdienen.

CPC

Schach für drei

Mit dem Programm "Schach für drei" aus Heft 4/5'92 habe ich Probleme. Bei den Befehlen |CLS und |SPRITE bekomme ich die Meldung "Unknown Command"! Ich habe schon alles probiert, bisher jedoch ohne Lösung.

Heinz Hofer, Wien

Zuerst die gute Nachricht: Bei uns läuft "Schach für drei" prima und ohne Fehler. Und nun die schlechte: Wenn dies bei Ihnen nicht so ist, stimmt offenbar etwas mit Ihrem abgetippten Programmexemplar nicht. "Unknown Command" wird nur dann ausgegeben, wenn die RSX-Erweiterung nicht initialisiert wurde. Haben Sie auch beide Data-Lader durchlaufen lassen? Befinden sich die Dateien CHESS.GRA und CHESS.MC auf der gleichen Diskette wie Ihr Basic-Hauptprogramm? Wenn ja, geben Sie einmal ein:

```
MEMORY &7FFF:
LOAD"CHESS.MC",&8000:
CALL &8000
```

Geben Sie anschließend bitte |CLS ein. Jetzt sollte der Befehl anerkannt werden. Sollte dies beim Ablauf des Haupt-Programms nicht der Fall sein, handelt es sich vermutlich um einen Tippfehler in der Zeile 230.
Falls alle Stricke reißen: Schicken Sie Ihr abgetipptes Listing auf einer Diskette zusammen mit einem frankierten Rückumschlag an die Redaktion zum Überprüfen.

CPC

Locomotive Basic für PC?

Ich habe mir aus beruflichen Gründen einen 286er AT zugelegt. Nun würde mich interessieren, ob es das Locomotive Basic oder ein ähnliches Basic auch auf dem PC gibt.

Michael Friedrich, Nürnberg

Leider müssen wir Sie enttäuschen. PC und CPC sind einander schon vom Speicherkonzept her so unähnlich, wie man es sich nur vorstellen kann. Allerdings können die fortschrittlicheren Basic-Dialekte auf dem PC (etwa Quick Basic, Power Basic oder auch das neue Visual Basic für Windows) eine ganze Menge. Was man beim CPC mit RSXen machen würde, wird beispielsweise bei Quick Basic als eigenständige SUBs definiert. Die PC-Basics bringen auch sehr ordentliche Grafikfunktionen mit – sie sind denen des CPC nur in keiner Weise ähnlich. Eine Umsetzung von Basic-Software vom CPC auf den PC ist nur dann denkbar, wenn:
– das entsprechende Programm lediglich die Standard-Basic-Syntax verwendet (Kontrollanweisungen, mathematische und trigonometrische Funktionen, logische Operatoren, Variablen, Felder, Stringfunktionen, DATAs und alphanumerische Bildschirm-PRINTs),
– weder Grafik noch POKEs, geschweige denn Maschinenroutinen im Programm vorkommen,
– alle Zugriffe auf Diskette oder Drucker in die Syntax des entsprechenden PC-Basic übersetzt werden.
Wer partout eine Brücke vom CPC zum PC bauen will, sollte dies lieber über Turbo Pascal versuchen. Die Version 3.0, die auf den CPCs unter CP/M läuft, ist seinerzeit auch für MS-DOS erschienen. Die Anpassung von Turbo-3.0-Programmen unter CP/M an den PC ist in der Regel kein Problem. Wen diese Dinge näher interessieren, der sollte einmal einen Blick in die Jahrgänge '87 und '88 der Zeitschrift PASCAL werfen. Restbestände davon lassen sich noch beim DMV-Verlag bestellen.

Wir sind ganz Ohr...

...für Ihre Fragen und Anregungen, was unser Heft betrifft – und alles andere rund um CPC oder PCW. Wollen Sie ein selbstgeschriebenes Programm im Heft unterbringen? Kennen Sie ein brandneues Programm, über das wir unbedingt etwas schreiben sollten? Kommen Sie mit einem der abgedruckten Listings nicht klar? Oder wollen Sie einfach mal mit einem von uns eine Runde fachsimpeln?

Dann ist unsere

Redaktions-Hotline

auch für Sie interessant.

Sie sind herzlich eingeladen – wir wissen zwar nicht auf jede Frage die richtige Antwort, aber mancher Knoten löst sich trotzdem schon am Telefon.

Jeden Dienstag von 16 bis 19 Uhr

hängt das komplette Redaktionsteam der CPC Amstrad International für Sie an der Leitung. Hier unsere Telefonnummern:



Jörg Gurowski (jg)
0 56 51 / 8 09 - 7 51



Ralf Schößler (rs)
0 56 51 / 8 09 - 7 52



Peter Schmitz (sz)
0 56 51 / 8 09 - 7 53

CPC

Demo-Bezugsadresse

Seit neuestem druckt ihr auch Demo-Tests ab. In der Ausgabe 8/9'92 wurde als Kontaktmöglichkeit eine Mailbox angegeben, mit der Nummer kann ich aber leider nichts anfangen! Können Sie mir eine Adresse nennen, wo ich die entsprechende Demo auf Diskette erhalten kann? Oder geben Sie mir einen Tip, wie ich die angegebene Mailboxnummer nutzen kann?

Tim Zylinski, München

Die Demos erhalten Sie auf Diskette unter anderem beim NoName-EDV-Service, Weidenweg 14, 2905 Edewecht. Die Mailboxnummer ist für all diejenigen interessant, die über eine RS-232-Schnittstelle (beispielsweise aus der Bauanleitung in Heft 6/7'92) und ein Modem sowie ein Terminal-Programm verfügen. (Ein solches Terminalprogramm wird es übrigens in Ausgabe 12/1'92/93 zum Abtippen geben.) Wenn der Rechner die betreffende Mailbox-Nummer zu den angegebenen Zeiten anwählt, läßt sich das gewünschte Demoprogramm kostenlos von dort "über die Leitung" holen.

CPC 464

Rekorderprobleme

Der eingebaute Datenrekorder meines CPC 464 scheint sich endgültig verabschiedet zu haben. Ist es möglich, einen gewöhnlichen Kassettenrekorder provisorisch mit dem Computer zu verbinden, so daß ich meine vorhandene Kassettensoftware wieder in den Speicher des Rechners bekomme und sie von dort aus auf Diskette retten kann?

Michaela Meier, Oberhausen

Es ist möglich. Wir können allerdings nicht für fehlerfreie Ladevorgänge garantieren. Sie müssen den (ausgeschalteten) Rechner zunächst öffnen. Dann suchen Sie auf der Rekorderplatine des CPC den IC 302. Pin 11 dieses IC verbinden Sie mit der Masseleitung am NF-Ausgang Ihres externen Kassettenrekorders (bei Klinkensteckern: dem "Fuß" des Steckers). Die Signalleitung (bei Klinkensteckern: die "Spitze" des Steckers) wird über einen Kondensator (4,7 Mikrofarad) mit Pin 6 des besagten IC verbunden.

Setzen Sie den genannten Kondensator bitte unbedingt in die Leitung zwischen Rekordersignal und Pin 6 des IC, um eine gleichstrommäßige Trennung zu gewährleisten.

Benutzen Sie als Signalquelle nach Möglichkeit einen unverstärkten Ausgang Ihres Kassettenrekorders (AUX, Amplifier, Radio). Wenn Sie einen Kopfhörerausgang verwenden, müssen Sie den Lautstärkeregler weit zurückstellen, damit der Kassettenrekorder dem CPC keinen zu hohen Pegel liefert.

PCW

Kalender aus Heft 2'90

Das von mir oft verwendete Kalenderprogramm zeigt seit der Wende für den Tag der deutschen Einheit ein falsches Datum an. Ohne Programmiererfahrung ist es mir jedoch gelungen, das Datum entsprechend zu ändern. Allerdings wurde mir dabei klar, daß dieser Feiertag nun als "beweglich" zu betrachten ist und außerdem nicht vor 1954 auftauchen dürfte. Die Berücksichtigung dieser Fakten und die eventuelle zusätzliche und zeitgerechte Einführung früherer Nationalfeiertage erfordert eine Programmerweiterung, die zu schreiben ich mir nicht zutraue. Hat Herr Herbert Tandler, der Autor dieses Programms, oder ein anderer User sich dieses Problems schon angenommen und eine entsprechende Erweiterung des Programms vorgestellt?

Josef Güntert, Taufkirchen

Wenn man nicht allzu großen Wert auf elegante Programmierung legt, lassen sich die betreffenden Änderungen (und auch andere Einschnitte im Kalendarium wie neue Festdaten oder die Streichung alter Feiertage) sehr einfach anbringen.

Wenn Sie folgende Zeilen ins Listing des Kalenderprogramms aus 2'90 einfügen, gibt es keine Schwierigkeiten mehr mit dem Tag der Deutschen Einheit. Zeile 3850 muß dann aus dem Originallisting gelöscht werden.

```
1360 IF fe$="ENDE" THEN
     1445
1445 IF jahr>1954 AND
     jahr<1991 THEN
     tag$(6,17)=" Tag
     der Dt. Einheit"
1446 IF jahr>1990 THEN
     tag$(10,3)=" Tag
     der Dt. Einheit "
3850 ' Zeile muß ge-
     löscht werden
```

Nachhall

zu Heft 8/9'92

Die letzte CPC war in mehrfacher Hinsicht ein Heft der Superlative. Leider auch im Hinblick auf einige ärgerliche Fehler, für die wir uns bei Ihnen entschuldigen und die wir hiermit korrigieren möchten.

Ein "dicker Hund" steckte im Bestückungsschema zur Lochrasterplatine des A/D-Wandlers (Artikel "Ohren für den CPC", Seite 45 unten). Der abgedruckte Schaltplan war allerdings in Ordnung. Die korrekte Anordnung der Bauelemente und Brücken haben wir nun hier abgebildet.

Der nächste Fehler hatte sich im Listing "Creator3.Ldr" des Rollenspiel-Artikels ("Begegnungen", Seite 63) versteckt. Im Artikel hieß es, das Binärfile "Creator3.Bin" werde nach dem Start des Basic-Laders erzeugt. Dieser POKEte auch brav seine Maschinencode-DATAs in den Speicher, dachte aber nicht daran, ein Binärfile zu erzeugen. Wie sollte er auch — wo wir doch den Sicherungsteil vergessen hatten?

Die Sache bekommt aber schnell Hand und Fuß, wenn man zwischen Zeile 220 und 230 von "Creator3.Ldr" die folgende Programmzeile einschiebt:

```
SAVE"creator3.bin",b,
&A000,&2A0
```

Der dritte Bug schließlich betrifft das Listing "Puzznix .Bas" ("Newton ist an allem schuld", Seite 27). Einige nicht druckbare Steuerzeichen in Zeile 1240, die sich durch die Überprüfung gemogelt hatten, brachten das Ausgabegerät durcheinander. Richtig muß die Zeile folgendermaßen aussehen:

```
1240 PEN 1:lev=1:sel=1:
strich$=CHR$(149)+CHR$(8)
+CHR$(10)+CHR$(149):
weg$=CHR$(32)+CHR$(8)
+CHR$(10)+CHR$(32):
LOCATE 15,sel*2+6:
PRINT strich$: GOSUB 2210
```

Wir werden uns große Mühe geben, solche Fehler in Zukunft zu vermeiden. Achtung, Züchter: Sollte jemand einen Spezialhund anzubieten haben, der versteckte Fehler in den Druckunterlagen einer Computerzeitschrift wittern und rechtzeitig vor Drucklegung aufspüren kann, so wären wir daran **sehr** interessiert.

Ihre CPC-Redaktion

Biete Software

FRAKTALGENERATOR 3D 30,- SOFTW EXP 20,- SPECIAL OFFER 1 45,- ZUS.90,- JAN DEPPISCH, WESERSTR. 18, 7550 RASTATT

Löse 3"-CPC-Games-Sammlung auf. Liste anfordern Tel. 0 41 05/7 61 94

Biete Joyce-Software: Fonts, dBase II, Wordstar 3.0, Multiplan, Spiele + viel Literatur. J.Niemeyer, Grünberger Str. 66, O-1035 Berlin, Fax/Tel. 5 88 16 40

Biete CPC 6128-Schneider-Joyce-Software: WordStar 3.0, dBase II, Multiplan, Spiele u. viel Literatur, alles gut erhalten. H.-J. Niemeyer, Grünberger Str. 66, O-1035 Berlin

*** cpc & joyce Public Domain Disks**
* Info geg. 2,-DM in Briefmarken anfordern:
* **PDI**, Pf 11 18, D-6464 Linsengericht **G**

FARB-Hardcopy für Star LC 10 Colour u.ä. und CPC für 30DM! M.Schuster Jakob-Kaiser-Weg 9 7888 Rheinfelden (Info gg.RP)

LOCOFONT 1 m. HANDB. 0 61 28/4 21 53

COPYSHOP; Anwendungspaket: STAR-MON, CREATOR-STAR u.a. K.Weisbrich Reddinger Weg 50 3053 Haste 0 57 23/8 23 38

Joyce: AMX-MaustInterface 50,-, Datenbank "Dateistar"+"Starmail", Kompatibel zu LocoScript 1, zus. f. 80,-, Desktop Publisher 50,-. Peter Werner, 0 30-3 44 47 77

Biete Hardware

CPC6128 incl. Farbmonitor, Joystick, Disketten mit div. Spielen und Programmen NUR 550 DM, Denis Bacina, Lothringerstr. 14, 85 Ngb. 40

CPC 664 Color, Vortex SP512 v. FD1 sowie versch. Spiele, Prgs. u. anderes. Liste anfordern bei Franz Feix, Tichling 17, 8311 Gottfrieding. Bitte frank. Rückumschlag beilegen

PCW512k, DRUCKER+5 1/4"-LW DEFEKT 40 3"DISK+MULTIPLAN, DR.GRAPH, DESKT-PUPL, MICA, COLOFONT ORIG+ DIV.DATABOX+DIV.LITERATUR ETC. GEBOT.V.KEMMLING, 3000 HANNOVER HÖPFNERSTR.2, TEL:05 11-41 69 83

Verkaufe CPC 6128+GT 65+EPSON LX 800 (9-Nadler)+7 Amstrad-Hefte+4 Leerdiscs+Software (u.a. Turbo-Pascal). Guter Zustand, wenig gebraucht. 500 DM. Tel.: 06 21/65 31 88

CPC 6128, CTM 644, DMP 2000, 5.25 LW, Cassettenstation, Bücher, Zeitschriften, Spiele, Anwendung und Databoxen usw!!! Nur komplett abzugeben für DM 450! Alex Maier/Schubertring 9/ 8522 Herzogenaurach

Verkaufe leicht defekten CPC 464 mit SP256, DD1, Turbo Pascal 3.0, Mica, Context Fraktal und CopyShop, 12 Bücher + Zeitschriften Drucker DMP2000 ca 700DM 0 73 61/6 22 68 ab 18Uhr

3"-DISKS ZU VERSCHERBELN (GEBR.) BOX MIT 42 Stck. FÜR 115 DM TEL. 05 21/1 51 10 (Heiko)

CPC6128+GT65+DMP2000+Word Star *Spiele VB 1000 DM Tel. 06 61/60 42 98

JOYCE: 512kB+3,5"-Floppy+Screen-Filter+Interface+SW-Flückiger Y., Mattenstr. 132, CH-2504 Biel

Suche Software

SUCHE KRAKOUT PRO FUER CPC 664/3" Werner Fischer Tel. 0 60 56/46 21

Suche Kassettenspiele für CPC 464 (Liste?) Tel. 0 78 43/26 45

Suche Hardware

Suche 9-NADEL-DRUCKER-SCHREIB-KOPF FÜR CMC CPA-80 GS(SHINWA) MODELL AV-80S2 NR.4008137 NEU od.GRBRAUCHT.KOPF-NR.BA45600A. J.WEINERT 5010 BERGHEIM 3 BUSSARDSTR.2 TEL.0 22 71/9 24 78

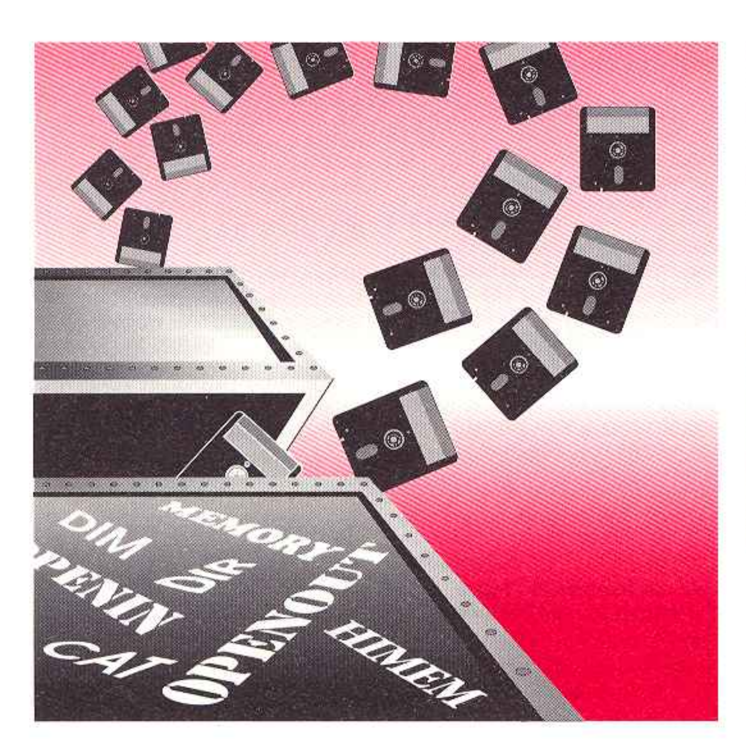
Serielle Schnittstelle, mögl. ECB und mit Treibersoftware K.Weisbrich Reddinger Weg 50 3053 Haste 0 57 23/8 23 38

SUCHE JOYCE-HARDWARE; ANGEBOTE AN W.WEISER KAZBOECKSTR. 2, 8900 AUGSBURG

Verschiedenes

Farbbänder f.JOYCE, PCW 9512, NLQ 401, DMP 2000 ab 9,- Etiketten, Disketten (3"DM 63) u.weit. EDV-Zubehör. Kostenl. Preislisten anfordern. W.Kuhn, EDV-Zubehör, Hessenstr. 7, 6340 Dillenburg 2, Tel.0 27 71/3 26 88 **G**

LocoLink, wer hilft mir mit Datenübertrag. PCW-PC? 0 30-3 44 47 77



Die Trickkiste

Buntgemischte Programmsammlungen

Jetzt endlich können wir voller Stolz sagen: "Es ist soweit. Die Trickkiste entspricht dem, was wir uns darunter vorgestellt haben." Die vielen Einsendungen unserer Leser machten es uns schwer, die eine der anderen vorzuziehen. Deshalb hier vorerst ein kleiner Einblick in das, was unsere Leser geleistet haben.

Druckerstatus

CPC 464/664/6128

Wenn man eigene Programme entwickelt, kommt es immer auf die Feinarbeit an. Soll doch das eigene Produkt nicht die gleichen Fehler aufweisen wie ein bereits vorhandenes aus anderer Feder.

Eine dieser Feinheiten ist die Druckerportabfrage. So ist der CPC in der Lage zu überprüfen, ob ein Drucker angeschlossen oder gerade außer Betrieb (offline) oder gar nicht erst angeschaltet ist.

Das kann manchem eine große Hilfe sein. Stellen Sie sich zum Beispiel vor, Sie wollen riesige Tabellen ausdrucken, haben den Ausdruck gestartet und ... es passiert nichts.

Haben Sie Ihre Daten bisher noch nicht abgespeichert, müssen Sie den Drucker nachträglich anschalten. Während dieses Vorgangs kommt es aber gelegentlich vor, daß der Drucker auf dem Weg wichtige Daten verschluckt oder daß sie gar verfälscht werden.

Aber das Programm läßt sich nicht beirren und schickt munter weitere Daten an den Drucker ab.

Warum also nicht die nachfolgende Routine einbauen, die locker in einer Programmzeile untergebracht werden kann? Sie überprüft den aktuellen Status des Druckers. Durch den im Sprachumfang des Locomotive-BASIC enthaltenen Befehl EVERY kann diese Routine während des Druckvorgangs alle paar Sekunden angesprungen werden. Ist der Drucker offline, kann entsprechend eine durch Tastendruck zu bestätigende Meldung ausgegeben werden.

Andreas Knösel/rs

```
100 ' Druckerstatus                              [1769]
110 ' CPC 464/664/6128                           [981]
120 ' (c) 1992 Andreas Knoesel                   [1299]
130 '          & CPC International               [2118]
140 status%=INP(&F500)                           [1396]
150 IF status%=&1A THEN PRINT"Drucker ist        [4738]
betriebsbereit!"
160 IF status%=&5A THEN PRINT"Drucker ist        [5574]
ausgeschaltet!"
```





Der Grüne Punkt

CPC 464(plus)/664/6128(plus)

Überall findet man inzwischen schon das "Duale System". Fast jede Konservendose, jede Glasflasche und jede Pappschachtel hat ihn schon, die Garantie für Wiederverwertung: den Grünen Punkt.

Jeder Computerbesitzer sollte sich jetzt einmal Gedanken über die Wiederverwertbarkeit seines Rechners machen.

Beim CPC gibt es so manches Teil, das beim totalen Defekt sinnvoll weiterverwendet werden könnte.

- Nachdem die "Innereien" entfernt und einige Abdichtarbeiten an der Tastatur durchgeführt worden sind, eignet sich dieser Teil wunderbar als Blumenbehältnis.
- Die Tasten vom Keyboard könnten gut in Kindergärten oder in Grundschulen verwendet werden. Gerade zum Erlernen von zusammenhängenden Buchstabenketten ein günstiges und zudem praktikables Lehrmittel.
- Sollte der Monitor noch funktionieren, kann er mit geringem Kostenaufwand in einen Fernseher verwandelt werden.
- Sämtliche Platinen werden sicher gerne von Bastlern abgenommen.

Rundum eine perfekte Sache, sollte man meinen. Warum also nicht auch dem CPC einen grünen Punkt zukommen lassen?

Da sich bestimmte Institutionen damit sicher etwas schwertun werden, sollten Sie das am besten in die eigene Hand nehmen.

Tippen Sie einfach das nachfolgende Programm ein, und speichern Sie den entstandenen Punkt ab. Eine beliebige Hardcopyroutine sorgt dann für den perfekten Ausdruck auf ihrem Drucker.

Selbstverständlich liegt die Größe des grünen Punktes in Ihrem Ermessen. So können auch andere entsorgungsleichte Geräte Ihren individuellen Punkt abbekommen.

Thomas Hombert/rs

```
100 ' Der Gruene Punkt                           [1838]
101 ' CPC 464(plus)/664/6128(plus)               [1940]
102 ' (c) 1992 Thomas Hombert                    [1542]
103 '         & CPC International                [2118]
104 INPUT"Durchmesser :",d:MODE 1:INK 0,15 [12685]
:INK 1,0:BORDER 15:e=1:f=1:g=320:h=200:i=d
:GOSUB 108:f=-1:e=2:GOSUB 108:f=1:h=200+d*
0.176:i=d*0.831:GOSUB 108:e=1:f=-1:h=200-d
*0.176:GOSUB 108:f=1:h=200+d*0.3802:i=d*0.
507:GOSUB 108:e=2:f=-1
105 h=200-d*0.3802:GOSUB 108:FOR a=0 TO d* [9858]
0.2459:j=d*0.3802:k=d*0.5027:l=d*0.085:m=d
*0.58:PLOT 320,200+j+d/2+a/4,2:DRAW 320+k,
200+j+l+a:PLOT 320,200+j-d/2-a/4:DRAW 320+
k,200+j-l-a:PLOT 320,200-j-d/2-a/4,1:DRAW
320-k,200-j-l-a
106 PLOT 320,200-j+d/2+a/4:DRAW 320-k,200- [11715]
j+l+a:NEXT:FOR a=1 TO 10 STEP 0.5:q=a/1.5:
PLOT 322,200+j-d/2+a,1:DRAW 320+m-q,200+j+
a/2:PLOT 322,200+j+d/2-a:DRAW 320+m-q,200+
j-a/2:PLOT 318,200-j+d/2-a,2:DRAW 320-m+q,
200-j-a/2:PLOT 318,200-j-d/2+a
107 DRAW 320-m+q,200-j+a/2:NEXT:END          [1894]
108 FOR a=0 TO i STEP 2:b=f*SQR(i^2-a*a):F [7753]
OR w=-1 TO 1 STEP 2:PLOT g+b,h+w*a,e:DRAWR
-2*b/2,0:NEXT:NEXT:RETURN
```

Bildeinblendung mit Komfort

CPC 464(plus)/664/6128(plus)

Als sichtbares Resultat unserer Trickkiste in der CPC 6/7'92 möchten wir Ihnen diese RSX-Erweiterung vorstellen. Animiert durch die Programme "Raindown" und "ScreenScroll", machte sich unser Leser Michael Müller daran, eine RSX-Erweiterung zu programmieren, die eine ganze Menge zu bieten hat. Wer also seine eigenen Programme erheblich aufpeppen möchte, sollte sich die nachfolgend erläuterten Befehle einmal genauer anschauen. Vorher seien jedoch noch kurz die Abkürzungen für eventuell zu übergebende Parameter erläutert:

"x", "y" geben jeweils die Koordinaten der oberen linken Ecke eines Bildschirmausschnitts an, "b" wiederum gibt die Breite und "h" die Höhe des Bildschirmausschnitts an. Alle Werte werden wie bei der Darstellung eines Fensters (WINDOW) in MODE 2 bearbeitet.

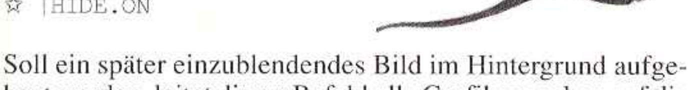
Doch nun zu den Befehlen:

☆ |HIDE.ON

Soll ein später einzublendendes Bild im Hintergrund aufgebaut werden, leitet dieser Befehl alle Grafikausgaben auf die unsichtbare Bildschirmseite um.

☆ |HIDE.OFF

Alle Bildschirmausgaben erscheinen wieder direkt auf dem Monitor.



☆ |DELAY,t

Verzögert die automatischen BLEND-Befehle um den Wert "t"; "t" entspricht zirka einer 1250stel Sekunde. Der Wert 255 ergibt also eine Verzögerung von einer Fünftelsekunde.

☆ |COPY,x,y,b,h

Hiermit kopieren Sie den angegebenen Bildausschnitt von der unsichtbaren zur sichbaren Seite. Durch Anwendung von Schleifen können mit diesem Befehl sehr viele interessante Effekte erzielt werden.

☆ |SPRAY

Der unsichtbare Bildschirm wird wie mit einer Spraydose über den sichtbaren Bildschirm gesprüht.

☆ |MOSAIC

Das unsichtbare Bild wird in Form von Mosaikbausteinen eingeblendet.

☆ |COMB,x,y,b,h

Der entsprechende Bildschirmausschnitt wird durch den anderen ersetzt. Dies sieht ungefähr so aus, als würden zwei Kämme mit ihren Zinken ineinander fahren.

☆ |RAIN.U,x,y,b,h / |RAIN.D,x,y,b,h

Der ausgewählte Bildbereich "regnet" von oben/unten auf den Monitor.

☆ |SLIDE.IN.L / SLIDE.IN.R

Von links beziehungsweise rechts schiebt sich der Bildschirmeinschnitt ein.

☆ |ROLL

Diese Funktion führt einen sogenannten Hardwarescroll aus, der die komplette Bildschirmseite "einrollen" läßt.

Insgesamt also jede Menge interessante Befehle. Zur Demonstration der genannten Funktionen steht mit dem zweiten Listing eine kleine Demoroutine zur Verfügung. Da sie auf ein Binärprogramm mit dem Namen TRICK3.BIN zugreift, muß dieses zuerst von dem ersten Listing generiert werden. Laden Sie also Listing eins ein, und löschen Sie den REMark-Befehl in der Zeile 200. Durch einmaligen Start wird jetzt die benötigte Binärdatei angelegt, und das Demonstrationsprogramm läuft reibungslos ab.

Michael Müller/rs

```
100 ' TRICK3.BAS (Initialisiert M-Code)      [2205]
110 ' CPC 464(plus)/664/6128(plus)            [1940]
120 ' (c) 1992 Michael Mueller                [1747]
130 '          & CPC International            [2118]
140 MODE 2:BORDER 0:INK 0,0:INK 1,13,26       [2130]
150 LOCATE 34,12:PRINT "Bitte warten"         [2225]
160 MEMORY &3FFF:a=0:b=0:c=&8000:a$=""        [1132]
170 FOR i=1 TO 28:READ a$:FOR j=1 TO LEN(a [1823]
$) STEP 2
180 a=VAL("&"+MID$(a$,j,2)):POKE c,a:b=b+a [3417]
:c=c+1:NEXT:NEXT
190 IF b<>112397 THEN CLS:CALL &BC02:PRINT [2387]
 "Fehler!":END
200 'SAVE "TRICK3.BIN",B,&8000,&3BB           [1419]
210 CALL &8000:|HIDE.ON:CLS:|HIDE.OFF         [2057]
220 CALL &BC02:MODE 2:END                     [941]
230 DATA 011380218580CDD1BCAF32BD833EC9320 [5868]
080C93980C38980C39E80C3A980C3CD80C3F6
240 DATA 80C32581C36C81C3DB81C33482C38482C [5082]
3BE82C30183484944452E4FCE484944452E4F
250 DATA 46C644454C41D9434F50D953505241D94 [3941]
D4F534149C3434F4DC25241494E2ED5524149
```

```
260 DATA 4E2EC4534C4944452E494E2ECC534C494 [5231]
4452E494E2ED2524F4CCC000000000000600CD
270 DATA 5F833E40CD08BC010CBCED490130BDED4 [4385]
9C90600CD5F833EC0CD08BCC9FE01C5C2A483
280 DATA C1DD7E0032BD83C93ABD83FE00C801780 [3294]
0110000210000EDB03DC2BD80C90604CD5F83
290 DATA DD4600CB20CB20CB20DD4E022ABB83CD1 [5272]
9BDC5E50600545DCBBCEDB0E1C1CD26BC10EF
300 DATA C90600CD5F830100402100C0CD19BD545 [4776]
DCBBA1A7757ED5F5F197CF6C0670B78B1C204
310 DATA 810100401100C0210040EDB0C90600CD5 [3430]
F830100102100C0CD19BDC5E50608545DCBBA
320 DATA 1A7713231A771B2B7CC6086710EEE1C15 [3409]
7ED5F5F1967E607F6C0677DE6FE6F0B78B1C2
330 DATA 3381010040110C0210040EDB0C90604C [2888]
D5F83DD46020E01CD19BDC5DD4600CB20CB20
340 DATA 2ABB83C5E5DD7E02910600545DCBBCD51 [4142]
6005F19D1EDB0E1C1CD26BCCD26BC10E2DD46
350 DATA 00CB20CB201608DD5E021D2ABB8319C5E [4363]
5DD7E02910600545DCBBCD516005FAFED52D1
360 DATA EDB8E1C1CD26BCCD26BC10E0CDB780C10 [3418]
C109CC90604CD5F83DD4600CB20CB20CB2005
370 DATA 05DD70002ABB83CD26BC10FBDD4600C5E [5349]
50602DD4E02CDE480E1C1C5E5C5E50600DD4E
380 DATA 02545DCD29BCEBEDB0E1C1CD29BC10EAC [4475]
DB780E1C1CD29BC10D2DD4600DD4E022ABB83
390 DATA CDE480C90604CD5F83DD4600CB20CB20C [4181]
B2005DD70002ABB83C5E50602DD4E02CDE480
400 DATA E1C1C5E5C5E50600DD4E02545DCD26BCE [3185]
BEDB0E1C1CD26BC10EACDB780E1C1CD26BC10
410 DATA D2DD4600DD4E022ABB83CDE480C90604C [3698]
D5F83DD46020E01C5DD4600CB20CB20CB202A
420 DATA BB83C5E5DD7E02910600545DCBBCD5160 [3319]
05F19D1EDB0E1C1CD26BC10E5CDB780C10C10
430 DATA D1C90604CD5F83DD46020E01C5DD4600C [3410]
B20CB20CB201600DD5E022ABB83192BC5E5DD
440 DATA 7E02910600545DCBBCD516005FAFED52D [2483]
1EDB8E1C1CD26BC10E3CDB780C10C10C8C906
450 DATA 00CD5F833E1ECD51830107BCED4904ED7 [3185]
9FE1AF220830106BCED4904ED793DFEFFC208
460 DATA 8306C80E502100C0CDE4803E00CD51830 [3434]
107BCED4904ED79FE1AF24A830106BCED4904
470 DATA ED793CFE1FC23283C901E803110000210 [3810]
000EDB0C319BDB8C2A48378FE00C8DD7E063D
480 DATA 67FE50F2A483DD7E043D6FFE19F2A483D [3829]
D7E023D84FE50F2A483DD7E003D85FE19F2A4
490 DATA 8316C05CD52600545D29291929292929D [3000]
11922BB83C9E1CD19BD01107FED49014B7FED
500 DATA 49CD19BD01547FED49C9               [1276]
```

```
100 ' TRICK3.DEM (Demonstriert Einblendung [3425]
)
110 ' CPC 464(plus)/664/6128(plus)         [1940]
120 ' (c) 1992 Michael Mueller             [1747]
130 '          & CPC International         [2118]
140 MODE 2:BORDER 0:INK 0,0:INK 1,13,26:LO [6438]
CATE 34,12:PRINT "Bitte warten"
150 MEMORY &3FFF:LOAD"trick3.bin",&8000:CA [2923]
LL &8000
160 |HIDE.ON:CLS:FOR i=1 TO 100:MOVE i,i:D [3242]
RAW RND*640,RND*400:NEXT:|HIDE.OFF
170 CLS:INK 1,26                           [815]
180 MOVE 0,0:DRAW 639,0:DRAW 639,398:DRAW  [3099]
0,398:DRAW 0,0
190 MOVE 207,0:DRAW 207,400:MOVE 208,0:DRA [2366]
W 208,400
200 MOVE 431,0:DRAW 431,400:MOVE 432,0:DRA [3162]
W 432,400
210 MOVE 0,126:DRAW 640,126:MOVE 0,128:DRA [2406]
W 640,128
220 MOVE 0,270:DRAW 640,270:MOVE 0,272:DRA [1787]
W 640,272
230 LOCATE 2,2:PRINT "COPY"                [1965]
240 LOCATE 28,2:PRINT "COPY"               [1383]
250 LOCATE 56,2:PRINT "COPY"               [1359]
260 LOCATE 2,10:PRINT "COPY"               [1977]
270 LOCATE 28,10:PRINT "COMB"              [1504]
280 LOCATE 56,10:PRINT "COPY"              [1501]
290 LOCATE 2,19:PRINT "COPY"               [2505]
300 LOCATE 28,19:PRINT "COPY"              [1251]
310 LOCATE 56,19:PRINT "COPY"              [1439]
320 GOSUB 730:FOR i=1 TO 26:|COPY,i,9,1,9: [2619]
GOSUB 740:NEXT
330 GOSUB 730:FOR i=80 TO 55 STEP -1:|COPY [4480]
,i,9,1,9:GOSUB 740:NEXT
340 GOSUB 730:FOR i=1 TO 8:|COPY,27,i,28,1 [2982]
:GOSUB 750:NEXT
350 GOSUB 730:FOR i=25 TO 18 STEP -1:|COPY [2583]
,27,i,28,1:GOSUB 750:NEXT
360 GOSUB 730:FOR i=1 TO 13:|COPY,i,1,1,8: [2228]
|COPY,27-i,1,1,8
370 GOSUB 740:NEXT                         [876]
380 GOSUB 730:FOR i=13 TO 1 STEP -1:|COPY, [4176]
54+i,18,1,8:|COPY,81-i,18,1,8
390 GOSUB 740:NEXT                         [876]
400 GOSUB 730:FOR i=1 TO 4:|COPY,55,i,26,1 [4551]
:|COPY,55,9-i,26,1
410 GOSUB 750:NEXT                         [940]
420 GOSUB 730:FOR i=4 TO 1 STEP -1:|COPY,1 [3647]
,17+i,26,1:|COPY,1,26-i,26,1
430 GOSUB 750:NEXT                         [940]
440 GOSUB 730:|DELAY,64:|COMB,27,9,28,9    [3825]
450 GOSUB 730:CLS                          [1046]
460 MOVE 0,0:DRAW 639,0:DRAW 639,398:DRAW  [3099]
0,398:DRAW 0,0
470 MOVE 319,0:DRAW 319,400:MOVE 320,0:DRA [1687]
W 320,400
480 MOVE 0,190:DRAW 320,190:MOVE 0,192:DRA [1880]
W 320,192
490 MOVE 320,206:DRAW 640,206:MOVE 320,208 [3036]
:DRAW 640,208
500 LOCATE 2,2:PRINT "RAIN.U"              [1866]
510 LOCATE 42,2:PRINT "SLIDE.IN.R"         [1763]
520 LOCATE 2,15:PRINT "SLIDE.IN.L"         [2014]
530 LOCATE 42,14:PRINT "RAIN.D"            [1443]
540 GOSUB 730:|DELAY,16:|RAIN.U,1,1,40,13  [2610]
550 GOSUB 730:|DELAY,16:|RAIN.D,41,13,40,1 [3208]
3
560 GOSUB 730:|DELAY,16:|SLIDE.IN.R,41,1,4 [3441]
0,12
570 GOSUB 730:|DELAY,16:|SLIDE.IN.L,1,14,4 [3222]
0,12
580 GOSUB 730:CLS                          [1046]
590 MOVE 0,0:DRAW 639,0:DRAW 639,398:DRAW  [3099]
0,398:DRAW 0,0
600 LOCATE 38,12:PRINT "SPRAY"             [1815]
610 GOSUB 730:|SPRAY                       [1484]
620 GOSUB 730:CLS                          [1046]
630 MOVE 0,0:DRAW 639,0:DRAW 639,398:DRAW  [3099]
0,398:DRAW 0,0
640 LOCATE 37,12:PRINT "MOSAIC"            [1113]
650 GOSUB 730:|MOSAIC                      [1829]
660 GOSUB 730:CLS                          [1046]
670 MOVE 0,0:DRAW 639,0:DRAW 639,398:DRAW  [3099]
0,398:DRAW 0,0
680 LOCATE 37,12:PRINT "ROLL"              [1145]
690 |HIDE.ON:LOCATE 30,12:PRINT "(c)CPC In [3916]
ternational":|HIDE.OFF
700 GOSUB 730:|ROLL                        [1768]
710 GOSUB 730:GOSUB 730                    [1204]
720 CALL &BC02:MODE 2:END                  [941]
730 FOR j=1 TO 1000:NEXT:RETURN            [1597]
740 FOR j=1 TO 25:NEXT:RETURN              [1683]
750 FOR j=1 TO 50:NEXT:RETURN              [1244]
```

Micro-Schrift

CPC 6128(plus)

Gerade in den Bildschirmmodi 1 und 0 wünscht man sich so manches Mal, eventuell doch mit einer anderen, platzsparenderen Schrift arbeiten zu können. Wären in Mode 0 anstelle der bisher nur 20 Zeichen pro Zeile 40 wünschenswert, hätte im Mode 1 eine 80-Zeichen-Darstellung große Vorteile.

Doch wie es nun einmal ist, nichts ist von Haus aus perfekt, kann jedoch nachträglich verbessert werden.

Zumindest auf dem CPC 6128 und 6128 Plus kann dieser Wunsch nun erfüllt werden. Alles, was Sie tun müssen, ist, das nachfolgende Listing einzugeben und bei Bedarf zu starten. Ab sofort liegen drei neue RSX-Kommandos zu Ihrer freien Verfügung bereit.

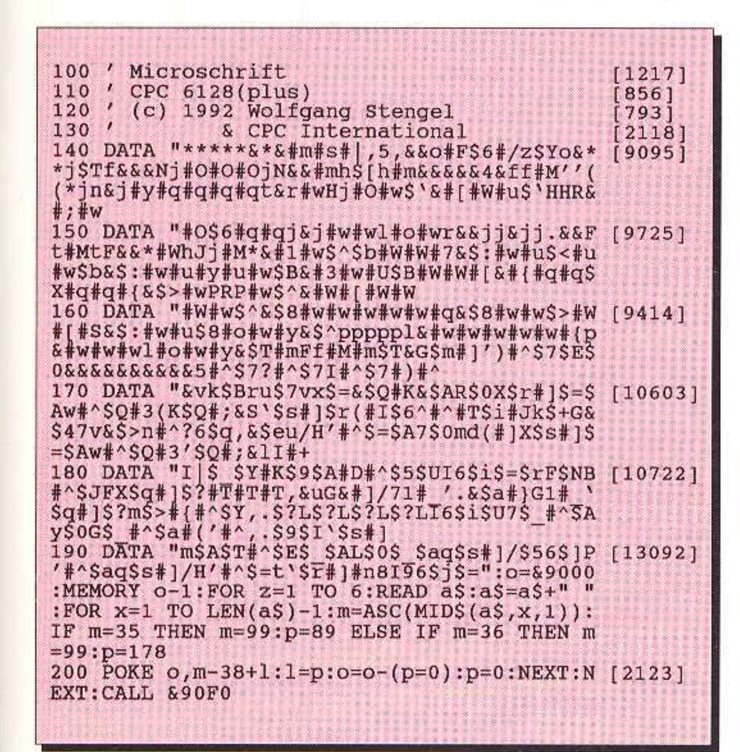
|PEN,x wählt die Farbe aus, mit der der Text auf den Bildschirm gebracht werden soll.

|LOC,x,y legt die X-/Y-Koordinaten für den Anfang der Zei-

chendarstellung fest. Die Koordinaten werden genau wie bei dem Original LOCATE-Befehl angegeben.

**|PRI,text$** schreibt den in der Variablen text$ gespeicherten Text auf den Bildschirm. Dabei sollten Sie darauf achten, daß diese Routine nur mit Großbuchstaben arbeitet. Alles andere wird als unverständliches Kauderwelsch geschrieben.

Wolfgang Stengel/rs



```
100 ' Microschrift                               [1217]
110 ' CPC 6128(plus)                             [856]
120 ' (c) 1992 Wolfgang Stengel                  [793]
130 '          & CPC International                [2118]
140 DATA "*****&*&#m#s#|,5,&&o#F$6#/z$Yo&* [9095]
*j$Tf&&&Nj#O#O#OjN&&#mh$[h#m&&&&&4&ff#M''(
(*jn&j#y#q#q#q#qt&r#wHj#O#w$`&#[#W#u$`HHR&
#;#w
150 DATA "#O$6#q#qj&j#w#wl#o#wr&&jj&jj.&&F [9725]
t#MtF&&*#WhJj#M*&#1#w$^$b#W#W#7&$:#w#u$<#u
#w$b&$:#w#u#y#u#w$B&#3#w#U$B#W#W#[&#{#q#q$
X#q#q#{&$>#wPRP#w$^&#W#[#W#W
160 DATA "#W#w$^&$8#w#w#w#w#w#q&$8#w#w$>#W [9414]
#[#S&$:#w#u$8#o#w#y&$^pppppl&#w#w#w#w#w#{p
&#w#w#wl#o#w#y&$T#mFf#M#m$T&G$m#]')#^$7$E$
0&&&&&&&&&&5#^$7?#^$7I#^$7#)#^
170 DATA "&vk$Bru$7vx$=&$Q#K&$AR$0X$r#]$=$ [10603]
Aw#^$Q#3(K$Q#;&S`$s#]$r(#I$6^#^#T$i#Jk$+G&
$47v&$>n#^?6$q,&$eu/H'#^$=$A7$0md(#]X$s#]$
=$Aw#^$Q#3'$Q#;&lI#+
180 DATA "I|$ $Y#K$9$A#D#^$5$UI6$i$=$rF$NB [10722]
#^$JFX$q#]$?#T#T#T,&uG&#]/71#_'.&$a#}G1#_`
$q#]$?m$>#{#^$Y,.$?L$?L$?L$?LI6$i$U7$_#^$A
y$0G$_#^$a#('#^,.$9$I`$s#]
190 DATA "m$A$T#^$E$_$AL$0$_$aq$s#]/$56$]P [13092]
'#^$aq$s#]/H'#^$=t`$r#]#n8I96$j$=":o=&9000
:MEMORY o-1:FOR z=1 TO 6:READ a$:a$=a$+" "
:FOR x=1 TO LEN(a$)-1:m=ASC(MID$(a$,x,1)):
IF m=35 THEN m=99:p=89 ELSE IF m=36 THEN m
=99:p=178
200 POKE o,m-38+l:l=p:o=o-(p=0):p=0:NEXT:N [2123]
EXT:CALL &90F0
```

Mini Diskettenmonitor

CPC 464(plus)/664/6128(plus)

Dieses Programm darf auf gar keinen Fall auf Ihrer Utility-Diskette fehlen. Erlaubt es doch bei einer bescheidenen Größe von nur 1 kByte, den kompletten Inhalt einer Diskette zu betrachten.

Alles, was Sie hierzu wissen müssen, ist, mit welchem Format die Diskette belegt ist. Nach dem Start meldet sich der Diskettenmonitor mit einem zweizeiligen Menü, das die momentan eingestellten Diskettenparameter sowie die zur Verfügung stehenden Kommandos anzeigt.

Wählen Sie jetzt die Funktion "(1)HPar" an, fragt das Programm nach zwei Werten, die durch Komma getrennt eingegeben werden müssen. Der erste Wert gibt hier das Laufwerk an, welches mit 0 für Laufwerk A und 1 für Laufwerk B bezeichnet wird.

Der zweite Parameter entscheidet nun, in welchem Format die Diskette formatiert ist. &42 steht hier für CP/M2.2 beziehungsweise CP/M3.0, &C1 für das CPC-DATA-Format, und &1 wählt ein CP/M86 oder MS-DOS-Diskettenformat an.

Nach dem Rücksprung ins Menü sollte jetzt Funktion "(2)DPar" gewählt werden. Hier geben Sie den anzuzeigenden Sektor (1-9 beziehungsweise 1-8 für CP/M86) sowie die entsprechende Spur (0-39) an.

"(3)Read" liest nun den ausgewählten Sektor ein, und "(4)Show" zeigt diesen jetzt auf dem Bildschirm an.

Soll der eingelesene Sektor auf eine andere Spur kopiert werden, wählen Sie noch einmal die Funktion "(2)DPar" an, um ihn dann mit "(5)Write" abzuspeichern.

Martin Lehr/rs



```
100 ' Sektor Editor                              [363]
110 ' CPC 464(+)/664/6128(+)                     [1386]
120 ' (c) 1992 Martin Lehr                       [741]
130 '          & CPC International               [2118]
140 MODE 2:MEMORY &1FFF:WINDOW#0,1,78,1,4        [3016]

150 GOSUB 210:buf=&2000:drv=0:sec=1:trk=0: [3572]
fmt=&C1
160 cmd=&84:PRINT"Drv ";drv;"  Fmt ";"&";H [4838]
EX$(fmt);"  Sec ";sec;"  Trk ";trk
170 PRINT"(1)HPar (2)DPar (3)Read (4)Show  [4258]
(5)Write ";
180 INPUT w:ON w GOSUB 280,270,200,240,190 [2302]
:CLS:GOTO 160
190 cmd=&85                                [273]
200 fsec=fmt+sec-1:CALL &A006,cmd:CALL &A0 [3577]
1D,buf,fsec,trk,drv:RETURN
210 s=0:READ d$:FOR i=0 TO &36:d=VAL("&"+M [3589]
ID$(d$,2*i+1,2)):POKE &A000+i,d:s=s+d:NEXT
 i
220 READ s$:IF VAL(s$)=s THEN RETURN ELSE  [3312]
PRINT"ERR":END
230 DATA 000000000000DD6E00DD66012203A0210 [7064]
3A0CDD4BC2200A0793202A0C9DD7E005FDD7E0257D
D7E044FDD6E06DD6607DF00A0D83205A0C9,&14FB
240 WINDOW #1,1,48,5,22:WINDOW #2,51,66,5, [2150]
22
250 FOR z=0 TO &1FF:r=PEEK(buf+z):a=r:IF ( [3608]
(a>127) OR (a<32)) THEN a=46
260 PRINT#1,HEX$(r,2);" ";:PRINT#2,CHR$(a) [2462]
;:NEXT z:RETURN
270 INPUT"(Sec),(Trk)";sec,trk:RETURN      [1494]
280 INPUT"(Drv),(Fmt)";drv,fmt:RETURN      [1430]
```

dBaseII-Dateien in eigenen Programmen

CPC 464(plus)/664/6128(plus)/PCW 8256/8512/9512

dBase, als immer noch am weitesten verbreitetes Datenbankprogramm für CPC und PCW, erlaubt es mit einem Minimum an Aufwand, beliebige Daten zu verwalten. Manchmal möchte der Programmierer jedoch aus eigenen Programmen heraus auf dBase-Daten zurückgreifen. Eine Beschreibung des dBase-Formats ist also nötig.

Eine dBase-Datei besteht grundsätzlich aus einem sogenannten Header und den eigentlichen Datensätzen. Im Header werden die Informationen wie Feldnamen und Feldlänge, die Anzahl der Datensätze und letztendlich das Datum des letzten Updates der Datenbank abgelegt. Hierzu sollten wir uns erst einmal folgende Tabelle anschauen:

Byte	Inhalt/Größe	Bemerkung
0	02 H	Bezeichnet dBase II-Datei
1-2	16 Bit Integer	Anzahl der Records
3-5	3 Bytes	Tag des letzten Updates
6-7	16 Bit Integer	Größe des Records
8-519	32 * 16 Byte Array	Feldbezeichner abgeschlossen mit 0DH
520	1 Byte	0D H, wenn alle 32 Felder benutzt werden, ansonsten 00 H

Nun fehlt nur noch eine Beschreibung der 32 Felder, die die jeweiligen Datensatzinformationen enthalten:

Byte	Inhalt/Größe	Bemerkung
0-10	11 Bytes	Feldname
11	1 Byte	Feldtyp
12	1 Byte	Feldlänge
13-14	16 Bit Integer	Felddatenadresse
15	1 Byte	Dezimalstellen bei numerischen Werten

Der Feldtyp wird durch C, N, L, D oder M angegeben. C steht dabei für Zeichen, N für numerisch, L für logisch, D für Datumsfeld und M für Memo.
Dem eigentlichen Datensatz ist ein Byte vorangestellt, in dem ein sogenanntes Löschkennzeichen eingetragen wird. Der Datensatz wird als gelöscht gekennzeichnet, indem dort der Wert 2AH (32D) eingetragen wird. Nicht gelöschte Sätze haben den Wert 20H (32D). Der Datensatz wird erst durch den Befehl Pack aus der Datenbank gelöscht.

Andreas Knösel/rs



JRT-Pascal und Leerstrings

CPC 6128(plus)/PCW 8256/8512/9512

JRT-Pascal, ein außerordentlich guter Public-Domain-Compiler, erfordert einige Einschränkungen in der Sprache, die man jedoch trickreich umgehen kann. Im Gegensatz zu Turbo Pascal kennt JRT-Pascal keine leeren Strings, was oft Probleme bereitet, wenn man zum Beispiel fertige Routinen in seine Programme einbinden möchte.
Da JRT-Pascal nur Strings mit einer maximalen Länge von 65535 Zeichen akzeptiert, wird die Länge als Integerzahl abgespeichert. Diese befindet sich in den ersten zwei Bytes eines Strings. Im nachfolgenden Listing wird ein varianter Record deklariert, der einen String der maximalen Länge enthält und einen Zugriff auf die Länge gestattet. Außerdem wird die Prozedur Delete durch DeleteStr ersetzt, damit Strings bis zum letzten Zeichen gelöscht werden können. Sonst muß nichts geändert werden, da alle weiteren Prozeduren von JRT-Pascal auch mit leeren Strings zurechtkommen.

Michael Haardt/rs

```
(* Leerstrings in JRT-Pascal  *)
(* (c) 1992 by Michael Haardt *)
(*        & CPC International *)

PROGRAM String_Test;

TYPE LongStr = RECORD
                CASE INTEGER OF
                  1: (str: STRING[65535]);
                  2: (mem: INTEGER);
                END;

VAR s1, s2, s3: STRING[20];
    i         : INTEGER;

PROCEDURE Nul(VAR s: LongStr);
BEGIN
 s.mem :=0;
END;

PROCEDURE DeleteStr(VAR s:LongStr; p1,p2:INTEGER);
BEGIN
IF (p1=1) AND (Length(s.str)=p2) THEN
 Nul(s)
ELSE
 Delete(s.str, p1, p2);
END;

BEGIN
 s1:='Test'; Nul(s2); s3:=s2;
 FOR i:=1 TO 4 DO BEGIN
  WriteLn(s1, Length(s1),' ',s2, Length(s2),' ',
          s3, Length(s3),' ',s1);
  DeleteStr(s1, 1, 1);
  s2:=Concat(s2,'!'); s3:=s2;
 END;
 WriteLn(s1, Length(s1),' ',s2, Length(s2),' ',
         s3, Length(s3),' ',s1);
END.
```



Directory unter Turbo-Pascal

CPC 6128(plus)/PCW 8256/8512/9512

Turbo-Pascal 3.0 Benutzer, die eigene Programme entwerfen, die auf beliebige Dateien zugreifen, können jetzt eine kleine DIR-Funktion einbinden. Sie erlaubt es, alle Dateien eines beliebigen Laufwerks auf den Bildschirm zu bringen.
Nach Eingabe und Start des Programms wird zuerst das Laufwerk abgefragt, dessen Dateien aufgelistet werden sollen. Hierbei darf jedoch nur der Kennbuchstabe, nicht jedoch — wie unter CP/M gewohnt — noch ein angehängter Doppelpunkt eingegeben werden.
Bevor die Routine aufgerufen wird, sollte das Hauptprogramm durch BDOS(45,255) das BDOS-Fehlerverhalten in den toleranten "return-mode" schalten. So führen falsche Laufwerksbezeichnungen, Tippfehler oder ähnliches nicht zum Absturz des ganzen Programms.
Natürlich kann diese Prozedur noch erweitert werden. Für Programme, die nur Dateien mit bestimmten Erweiterungen auflisten sollen (zum Beispiel PAS), kann in der FCB-Zuweisung anstelle der letzten drei Fragezeichen die Erweiterung eingegeben werden.
Denkbar ist auch eine Erweiterung der Ausgabefunktion. Zu Dokumentationszwecken wäre auch der Ausdruck auf den Drucker möglich. Die WRITE- und WRITELN-Anweisungen sind dann entsprechend durch ein LST zu ergänzen. Soll das Ausgabegerät variabel bleiben, ist auch eine Ergänzung in anderer Form denkbar. So muß das Ausgabegerät (LST, CON oder AUX) direkt beim Aufruf von ShowDir übergeben werden. Die Prozedur wird dann einfach mit "PROCEDURE ShowDir (VAR f: TEXT) deklariert. Der Aufruf erfolgt dann mit "ShowDir(LST) für die Druckerausgabe.
Ausgeben der Daten in eine Datei ist jedoch nicht ohne Zwischenspeicherung der Daten möglich, da sonst die FCBs durcheinander kommen würden. Bei der Verwendung von Balkenmenüs zur Auswahl der Dateien muß ebenfalls zwischengespeichert werden. Hier ist es dann auch angebracht, die gelesenen Daten zu sortieren.

Andreas Knösel/rs

```
PROCEDURE ShowDir;     { Zeigt Disketteninhalt an }

VAR FCB: ARRAY[0..35] OF CHAR;
                               { File Control Block }
    DMA: ARRAY[0..3,0..31] OF CHAR;
                        { Direct Memory Access Array }
    i,x: BYTE;                     { Hilfsvariable }
    s  : STRING[1];              { Eingabe-String }

BEGIN
 BDOS(26,ADDR(DMA));         { DMA-Adresse setzen }
 WRITE('DRIVE: ');
 READLN(s);             { LW-Bezeichnung abfragen }
 FCB:='-???????????                             ';
                          { FCB setzen, 36 Bytes }
 IF s<> '' THEN
  FCB[0]:=s[1]          { LW-Bezeichnung einsetzen }
 ELSE
  FCB[0]:='@';  { leer fuer derzeitiges Laufwerk }
 x:=BDOS(17,ADDR(FCB));
                     { Nach erstem Eintrag suchen }
 IF x < 4 THEN
  FOR i:=1 TO 11 DO WRITE(DMA[x,i])
 ELSE
  WRITE('No File');
 WRITELN;
 WHILE x< 4 DO
 BEGIN
  x:=BDOS(18); { Nach weiteren Eintraegen suchen }
  IF x<4 THEN
   FOR i:=1 TO 11 DO WRITE(DMA[x,i]);
  WRITELN;
 END;
END;

BEGIN;
ShowDir;
END.
```

Amsdos Update



CPC 464

Das Amsdos der CPC-Rechner bietet so manche Funktion, die im Detail noch ein wenig ausgefeilter werden könnte. Sieht man sich zum Beispiel den CAT-Befehl an, merkt man schnell, daß da noch etwas mehr zu gebrauchen wäre. Was würden Sie zum Beispiel davon halten, wenn Ihnen nicht nur stur die Dateinamen sowie der freie Speicherplatz einer Diskette angezeigt würde? Als Ergänzung möchten wir Ihnen nachfolgendes Listing anbieten. Es ergänzt die Systemroutine, die für den CAT-Befehl zuständig ist, dahingehend, daß nun auch gleich noch das ermittelte Diskettenformat sowie die Anzahl der Dateien im derzeit angewählten Userbereich angezeigt werden.

Doch zunächst sollten Sie das nachfolgende Listing abtippten. Nach dem Start wird die Datei AMSDATE.BIN angelegt, die mittels RUN"AMSDATE.BIN" gestartet werden kann. Nun ist die Erweiterung eingebunden.

Oliver Copp/rs

```
100 ' AMSDOS Update                          [1511]
110 ' CPC 464(plus)                          [779]
120 ' (c) 1992 Oliver Copp                   [608]
130 '        & CPC International             [2118]
140 DEFINT a-z:FOR a=0 TO 4:cs!=0:READ n$: [10702]
FOR b=0 TO 76:POKE &AF00+a*77+b,VAL("&"+MI
D$(n$,b*2+1,2)):cs!=cs!+ASC(MID$(n$,b+1,1)
)*(VAL("&"+MID$(n$,b*2+1,2))+1):NEXT:READ
c!:IF c!<>cs!THEN PRINT"Error in"a*10+90:E
ND
150 NEXT:SAVE"amsdate.bin",b,&AF00,&181,&A [2641]
F08
160 DATA 2140B0CDDDAF182F2110AF0EFFC316BDC [8999]
DCBBCCD37AFCD00B9CD06B9212F07CDDDAF216D06C
DDDAF219306CDDDAF2142B0CDDDAFC312C0219BBC3
6C3214CAF229CBC3E053266BEAF3278BEC9F5
170 DATA 600531                              [277]
180 DATA E5CDB4AFE1F1D5DF6EB0CDC5AF7EE6F0F [8972]
E402849FEC02840B72847210CB0CDDDAFE1182A3E2
CCD5ABBE106007EFEFF200704110E001918F478CDE
6AF2171B0CDDDAF3A01A7CDE6AF216BB0CDDD
190 DATA 651976                              [528]
200 DATA AF3E053266BEAF3278BEC92127B018262 [10830]
139B018212135B0181C2178BE36FF3E013266BECDC
5AF3ED077C93A00A7219FA8B7C82EDFC9E5CDDAAFE
1CDDDAF1897212CB07E23B7C45ABB20F8C90E
210 DATA 532169                              [379]
220 DATA 640600CDF8AF0E0A0600CDF8AF47C303B [10263]
0B9D8B9DA03B00491C3FAAFF578C630CD5ABBF1C90
A4E6F206469726563746F72792061766696C61626
C650A0A0D004441544100464F524D41543A20
230 DATA 478401                              [484]
240 DATA 0049424D0056454E444F5200043220414 [8928]
D53444F53205550444154452020284329203139393
2206279204F6C6976657220436F70700A0D0A0D001
3D5072066696C6573206F6E20757365722000
250 DATA 272389                              [493]
```

Multiface-Pokes

CPC 464(plus)/664/6128(plus)

Eine wahre Poke-Flut für aktuelle aber auch für klassische Spiele erreichte uns in letzter Zeit. Da selbstverständlich auch Spiele in der CPC einen hohen Stellenwert einnehmen sollen, wollen wir Ihnen diese Pokesammlung vorstellen. Doch zunächst einmal die Reihenfolge für die Eingabe der Pokes mit einem Multiface-II.

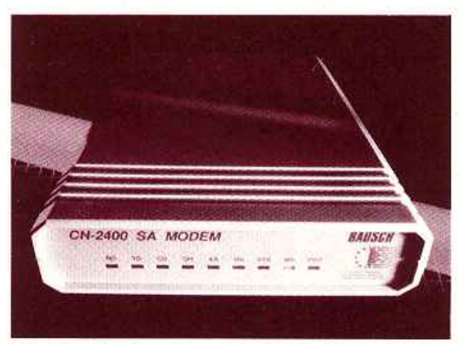
1. Roten Knopf am Multiface drücken und mittels <T> die Funktion Tool anwählen.
2. Da die nachfolgenden Pokes hexadezimal angegeben sind, muß mit <H> auf hexadezimal umgeschaltet werden.
3. <SPACE> schaltet nun auch Eingabe um.
4. Tippen Sie die angegebene Adresse sowie nachfolgend ihren Inhalt ein.
5. <RETURN> schließt die Eingabe ab.
6. Sollen weitere Pokes eingegeben werden, springen Sie zurück zu "3".
7. <ESC> springt in das Menü und <R> geht wieder zurück ins Programm.

Programm	Disk/Tape	Adresse	Poke
Black Tiger	D	5612	00
Bomb Jack 1	D	19FD	00
Bomb Jack 2	D	18E5	18
Cabal	D	0B1A	00
		1972	00
Chicago 30's	T	43DD	00
Dominator	D	09A4	FF
Dragon's Lair II	T	7D3D	00
Elevator Action	D	04A3	00
Flimbos Quest	D	7724	00
Hobgoblins	D	1234	00
Hotrod	D	4B2B	00
		4B2C	00
		4B2D	00
		4B36	00
IK+	D	29B7	??
		27C7	00
Marsport	D	7389	86
Monty Python	D	8B91	FF
		A388	00
Ninja Spirit	D	522D	00
North And South	D	0443	00
		13B2	??
Overlander	T	1815	00
Platoon	D	0135	01
Rastan Saga	D	0D30	FF
		05E0	??
Shinobi	D	15B7	65
		1407	00
Slap Fight	T	4F32	FF
Smash TV	D	1D5C	00
Spitting Images	D	36F3	C1
Strider	D	2AC2	00
		2BE9	00
The Simpsons	D	15AC	FF
		3B0E	00
Victory Road	T	3B5E	FF
		3B6F	FF
Wonder Boy	T	52E5	00

Computer auf Kontaktsuche

Modems an CPC und PCW

Daß auch dem CPC- oder PCWler alle Türen offenstehen, Modems und Mailboxen also nicht nur PC- oder Amiga-Usern vorbehalten sind, haben wir in den vergangenen Monaten mit unserem DFÜ-Kurs gezeigt. Was muß jedoch beachtet werden, was wird benötigt, um loslegen zu können?



Ein Standardmodem, das im Betrieb mit dem CPC keine Schwierigkeiten macht

Sowohl CPC als auch Joyce benötigen für die Kommunikation über das Telefonnetz neben einem Modem noch Zusatzhardware, die die Verbindung zwischen Rechner und Modem übernimmt. Da beide Computer nicht von Haus aus über eine serielle Schnittstelle verfügen, muß diese nachgerüstet werden. Beim PCW fällt die Wahl leicht, gibt es doch nur die CPS8256, die mit rund 200 DM oder bei gleichzeitigem Speicheraufbau entsprechend mehr zu Buche schlägt.

Die Qual der Wahl

Beim CPC hat man hier schon eine größere Auswahl. Von der original Schneider/Amstrad-Schnittstelle über die Amsoft und Philosoft bis hin zu unserem Schnittstellenselbstbau in der CPC 6/7'92 gibt es hier mehr als genug. Doch ausnahmsweise sollte beim CPC diesmal eine Regel mißachtet werden. Kauft man im Normalfall immer zuerst die Hardware, um dann die entsprechende Software anzuschaffen, muß man diesmal den Spieß umdrehen. So sollte man sich an einschlägig bekannte PD-Anbieter wenden, um von dort erst einmal ein paar DFÜ-Programme zu beziehen.
Selbstverständlich kann der Joycer auch diesen Weg gehen, um so gegebenenfalls doch ein besseres Programm als das mitgelieferte MAIL232.COM von der Systemdiskette zu finden.
Ist erst einmal die Anleitung der Software studiert, kann man schnell seine Wahl treffen.

Und noch etwas

Hat man sich für eine Schnittstelle und die entsprechende Software entschieden, folgt im nächsten Schritt die Suche nach einem passenden Modem. Da CPC und PCW bei Übertragungsgeschwindigkeiten von über 2400 Baud stark ins Holpern kommen, sollte ein Modem mit eben dieser maximalen Geschwindigkeit angeschafft werden. Modems dieser Geschwindigkeit sind je nach Ausbaustufe – sprich nach Extras und eventueller Postzulassung – für 150 bis 500 DM erhältlich.
Um nicht die Katze im Sack zu kaufen und eventuell für Zusatzfeatures, die weder CPC noch PCW ausnutzen können, bare Münze zu verschwenden, sollten Sie sich schon ein wenig mit der Materie befassen und eventuell nachfolgende Ratschläge genauer studieren.

V23 – Btx zum Ortstarif

Bei den Modempreisen schlägt sich besonders ein bestimmtes Feature nieder. Es handelt sich um die V23-Norm, die den Btx-Betrieb mit 1200/75 bps (Bytes pro Sekunde) erlaubt. Für CPC-Besitzer uninteressant, kann diese Norm für PCWler wichtig sein. Plant man zum Beispiel den zusätzlichen Kauf eines Btx-Moduls, kann so der Btx-Betrieb zum Ortstarif in Verbindung mit diesem Modem aufgenommen werden. Eine zusätzliche DBT03-Anschlußbox entfällt also.
Bei schlechten Telefonleitungen sind auch Fehlerprotokolle empfehlenswert. Haben Sie schon beim Telefonieren mit Freunden immer ein unliebsames Rauschen in der Leitung, sollte das Modem über die Standards MNP-4 oder MNP-10 verfügen. Es kann auch interessant sein, ein Modem mit integrierter Datenkomprimierung anzuschaffen. MNP-5 zum Beispiel kann bei reinen Textdateien eine Geschwindigkeitssteigerung von bis zu 100 % erreichen.
Will man mit anderen 8-Bit-Systemen Kontakt aufnehmen, ist es ebenfalls ratsam, die inzwischen nur noch selten verwendeten Normen Bell 103/V.21 und Bell 212/V.22 in die Kaufentscheidung einzubeziehen. Diese Normen stehen für den Datenaustausch mit 300 beziehungsweise 1200 bps.
Beachtet man all diese Punkte, kann eigentlich nichts mehr schiefgehen, und das Modem kann angeschafft werden.

Anschluß mit Tücken

Steht das Modem nun endlich zu Hause auf dem Schreibtisch, ist die Enttäuschung oftmals groß. Sind doch alle Kabel vorhanden, nur leider paßt das Telefonkabel nicht.
Hier kann aber Abhilfe geschaffen werden. Da fast alle Modems an der Rückseite über einen Western-Anschluß (amerikanische Telefonsteckernorm) verfügen, gibt es im Fachhandel schon fertige Kabel, die je nach Länge zwischen 5 und 20 DM kosten.
Stehen Telefon und Modem im gleichen Raum, bietet es sich an, eine neue Telefondose für rund 10 DM zu installieren. Da hier nur zwei beziehungsweise vier Kabel angeschlossen werden müssen, kann die Dose selbst installiert werden.
Hierzu schrauben Sie die alte Dose aus der Wand und lösen die zwei angeschlossenen Kabel von den Verbindern 1 und 2. Jetzt schließen Sie die zwei Kabel an der neuen Doppeldose an. Je nach Bauart müssen nun noch zwei Strippen von Pin 1 und 2 des ersten Steckerausgangs an Pin 1 und 2 des zweiten Ausgangs gelegt werden. Die Dose kann wieder an der Wand montiert und sowohl Telefon als auch Modem betrieben werden, und das, ohne immer wieder die Telefonstecker austauschen zu müssen.
Jetzt, ja jetzt kann endlich die erste Verbindung mit einer Mailbox hergestellt werden. Doch dazu im nächsten Heft mehr.

Ralf Schößler

Impressum

Herausgeber
Helmuth Schmitz
Chefredakteur (verantw.)
Peter Schmitz (sz)
Redaktion
Jörg Gurowski (jg), Ralf Schößler (rs)
Autoren dieser Ausgabe
Mike Behrendt, Hubert Cronenbrock, Rainer Dombrowski, Matthias Fink, Michael Haardt, Thomas Hombert, Dieter Kinzinger, Andreas Knösel, Günter Kramer, Markus Kräutner, Michael Kruse, Martin Lehr, Andreas Lober, Ivo Mieskes, Markus Müller, Michael Müller, Reiner Müller, Markus Satte, Ulrich Schmitz, Wolfgang Stengel, Marc Stiegemeyer, William Wiebrecht
Redaktions-Assistenz
Carmen Strube (cs)
Schlußredaktion
Marita Gleim, Dagmar Wilhelm-Ballhaus
Bereichsleitung
Uwe Siebert (Produktion)
Margarete Schenk, Helmut Skoupy (Montage/Reprografie)
Layout
Lars Völke
Fotografie
Petra Hugo, Bianca Krück
Fotosatz
Marcus Geppert, Andrea Kloss, Regina Sieberheyn
Montage/Reprografie
Monika Martin, Andrea Herschelmann, Susanne Eckhardt
Werbegestaltung
Mohamed Hawa
Anzeigenverkauf für PLZ 1, 4, 5
Gerlinde Rachow, Telefon: (0 56 51) 80 93 90
Sylvia Stephani, Telefon: (0 56 51) 80 93 80
Karina Ehrlich, Telefon: (0 56 51) 80 93 71
Bernd Heckmann, Telefon: (0 56 51) 80 93 81
Telefax: (0 56 51) 80 94 44
Anzeigenverkauf für PLZ 2 + 3
DMV-Verlagsbüro Hamburg,
Kunaustraße 4a, 2000 Hamburg 65
Leitung: Sylvia Ehrenpfordt
Anzeigenverkauf: Sabine Bindseil
Telefon: (0 40) 600 30 75-77, Telefax: (0 40) 601 54 75

Anzeigenverkauf für PLZ 6 - 8
DMV-Verlagsbüro München, Zaunkönigsweg 2c, 8000 München 82
Telefon: (0 89) 4 39 10 87, Telefax: (0 89) 4 39 10 80
Leitung: Britta Fiebig
Anzeigenverkauf: Peter Schätzle, Hannelore David, Ilona Sehm
Anzeigenverwaltung
Andrea Giese, Beate Kranz
Druckunterlagen-Disposition
Christina Wabra, Telefon: (0 56 51) 80 93 72
Anschrift Verlag/Redaktion
DMV Daten- und Medien-Verlag, Widuch GmbH & Co. KG
Fuldaer Straße 6, 3440 Eschwege
Telefon: (0 56 51) 8 09-0, Telefax: (0 56 51) 80 93 33
Vertrieb
Verlagsunion Erich Pabel– Arthur Moewig KG (VPM)
Friedrich-Bergius-Straße 20, 6200 Wiesbaden
Druck
Druckerei Jungfer, 3420 Herzberg
Bezugspreise
»CPC Amstrad International« erscheint zweimonatlich.
Einzelpreis DM 6,–/sfr. 6,–/öS 50,–
Abonnementpreise
Die Preise verstehen sich grundsätzlich einschließlich Porto und Verpackung.
Inland:
12 Ausgaben: DM 66,– 6 Ausgaben: DM 33,–
Europäisches Ausland:
12 Ausgaben: DM 96,– 6 Ausgaben: DM 48,–
Außereuropäisches Ausland:
12 Ausgaben: DM 120,– 6 Ausgaben: DM 60,–
Bankverbindungen:
Postscheck Frankfurt/M: Kto.-Nr. 230 43–608
Raiffeisenbank Eschwege: BLZ: 522 603 85, Kto.-Nr. 245 7008
Die Abonnementbestellung kann innerhalb einer Woche nach Auftrag beim DMV-Verlag, Postfach 250, 3440 Eschwege, schriftlich widerrufen werden. Zur Wahrung der Frist reicht der Poststempel. Das Abonnement verlängert sich automatisch um 6 Ausgaben, wenn es nicht mindestens 6 Wochen vor Ablauf beim Verlag schriftlich gekündigt wird. Für unverlangt eingesandte Manuskripte und Datenträger sowie Fotos übernimmt der Verlag keine Haftung. Die Zustimmung zum Abdruck wird vorausgesetzt.

ISSN der Deutschen Bibliothek: 0941-9217

Händlerverzeichnis

Gelsenkirchen

LYNX VERSAND
Udo Oestreich
TEL. 0209 20 70 22

Herne 2/Wanne-Eickel

BROSTEDT Computertechnik
HARDWARE – SOFTWARE
ENTWICKLUNG – VERTRIEB
Großes CPC-Sortiment: Über 600 Titel lieferbar – Spiele, Anwendungen, Public Domain – Wir vermarkten auch Ihre Programme – Liste anfordern!
Harkortstr. 54, W-4690 Herne 2, Tel.: 0 23 25/6 08 65

Eintragungen im Händlerverzeichnis, nach Städten geordnet, kosten je mm Höhe 6,– DM bei einer Spaltenbreite von 58 mm.

Köln

MB–Versand – Alles für den CPC!
Ihr Spezialist in Sachen Software & Hardware, PD & Demo – Kopierservice und Reparaturen
PF 501132, 5000 Köln 50, Tel.: 02236/47108
Mailboxservice (300-2400, 8, N, 1): 02236/83007

NoName EDV-Service Gbr
* Software * Hardware * Modems * Zubehör *
Postfach 50 11 32, 5000 Köln 50
Tel: 02236/47108 - FAX: 02236/83007

Löhne/Ostwestfalen

RADIO
Computer- & Softwarezentrum für Norddeutschland:
AMSTRAD-, SCHNEIDER- & VORTEX Regionalhändler & **SERVICE-CENTRALE.** Sämtl. Computer, Drucker, Peripherie & Zubehör v. A–Z, EDV Papier etc. • Discs
Fritz OBERMEIER COMPUTER • TELEFAX • BTX • HIFI • VIDEO • TV • + NEC • EPSON • TANDON • BROTHER • SEIKO • OKI • STAR • LOCO • etc.
am Bahnhof • Bünder Straße 20 • 4972 LÖHNE 1 • Tel. 0 57 32 61 26/32 46

Ratzeburg

NoName EDV-Service Gbr
* Software * Hardware * Modems * Zubehör *
z.Hd. W.Noisternig, Friedrich-Ebert-Str. 27
2418 Ratzeburg, Fax: 04541/84944

Nähere Informationen:

DMV-Verlag
Sylvia Stephani
Telefon (0 56 51) 8 09-3 80



Action, Fantasy und die Hits von gestern

Neue Spiele und Programmsammlungen

Der heiße Sommer verabschiedet sich und macht Platz für einen vielleicht nicht minder heißen Herbst. Aus den Ferien zurückgekehrt, wird der CPCler sogleich mit einer Flut neuer Spiele überrascht. Auch wer erst seit kurzem den Markt verfolgt, bekommt jetzt Gelegenheit, die großen Hits der vergangenen Jahre zu genießen: Neu erscheinende Spielesammlungen bieten die größten Erfolge jetzt gebündelt zu niedrigen Preisen. Abwechslung ist Trumpf: Wir haben uns für Sie unter anderem eine anspruchsvolle Simulation, Managementspiele, Action-Knaller und auch ein Lernspiel für die ganz Kleinen angeschaut.

Die meisten englischen Hersteller von Vollpreisspielen ziehen sich vom CPC-Markt zurück oder verringern ihren Ausstoß an neuer Software. Trost kommt aus Frankreich – oder von den vielen Spiele-Sammelpacks und Wiederveröffentlichungen. Auf diese Art und Weise kommt man oft preiswert an eine ganze Reihe guter Spiele.

SAINT DRAGON auf dem **MAX**-Pack von US Gold gehört leider nicht dazu. Auffallendste Merkmale sind eine schlecht gezeichnete Grafik, die von einem eintönigen Sound unterstützt und mit einem wenig originellen Spielprinzip zu absoluter Mittelmäßigkeit verbunden wird. Ach ja, bevor wir's vergessen: Es geht darum, sich mit einem Raumschiff – oh, Entschuldigung, einem Drachen – durch eine horizontal scrollende Landschaft zu kämpfen. Irgendwie bekannt? Nicht doch ...

Ein ungleiches Quartett

Auch **SWIV**, ein weiterer Bestandteil des MAX-Packs, ist nicht gerade das, was man gemeinhin als Hit bezeichnet. Diesmal scrollt das Spielfeld vertikal, die Grafik ist zwar sauber gestaltet, erhält aber erst durch die enorme Farbvielfalt von Schwarz und Gelb ihren besonderen Reiz. Ein oder zwei Spieler können mit einem Jeep und einem Hubschrauber gegen die Übermacht ankämpfen. Anfangs wird der Spieler gnadenlos unterfordert. Mit der Zeit und mit steigendem Schwierigkeitsgrad werden die Feinde jedoch origineller und abwechslungsreicher und das Spiel spannender. Jetzt kommen wir zur Sonnenseite der Packung. Von **TURRICAN II** (und seinem Vorgänger TURRICAN) habe ich bereits genug geschwärmt. Aber es ist nun einmal ein ungeheuer abwechslungsreiches, grafisch und programmtechnisch hochkarätiges Action-Spiel.

Bleibt noch **NIGHT SHIFT**. Auch dieses Programm haben wir erst vor kurzem vorgestellt. Noch mal in Kürze: In einer Spielzeugfabrik muß eine riesige Maschine überwacht werden, die zu einigen kleinen Defekten neigt. Ballern ist – ausnahmsweise einmal – nicht möglich. Trotzdem geht es heiß her. In höheren Levels ändert sich die Grafik kaum, das Spiel bekommt aber immer vertracktere Elemente. Ein eigenwilliges Spielprinzip hat diesem Spiel viele Freunde beschert – vollkommen zu Recht!

Wer NIGHT SHIFT und TURRICAN II noch nicht besitzt, sollte schnell zugreifen, SWIV als Bonus ansehen und SAINT DRAGON, na ja ... man braucht's ja nicht zu spielen.

MAX (MAXIMUM ACTION XTRA) 1) 2) 3) 4)

Hersteller: US Gold
Steuerung: Joystick oder Tastatur
Voraussetzung: für alle CPCs
Preis: etwa 75 DM
Anleitung: deutsch, englisch, französisch
Bemerkung: Turrican II ist vor kurzem auch als Billigspiel erschienen.

Bewertung:

SAINT DRAGON: 4	SWIV: 3
TURRICAN II: 1	NIGHT SHIFT: 1

Vor ein paar Jahren sorgte auf dem C-64 ein Programm namens **URIDIUM** kräftig für Furore. Ein grafisch blitzsauberes Spiel hatte Programmierer Andrew Braybrook da hingelegt. Auch spielerisch hob es sich angenehm von der Massenware ab.

Wiedergeburt einer Legende

Die Zeit verging, und die Computerwelt wartete gespannt auf die Ankündigung einer CPC-Version. Nichts geschah. Als man sich nach über einem Jahr bei Hewson dann dazu entschloß, das tapfere Schneiderlein mit einer Umsetzung zu bedenken, war es fast zu spät. Das Programm erschien von der Presse weitgehend unbeachtet und verkaufte sich mittelmäßig.

Jetzt ist die zweite Chance für URIDIUM gekommen, denn das Spiel liegt wiederveröffentlicht als Billigspiel vor.
Die Story um eine Bedrohung des Sonnensystems durch bösartige Wesen wollen wir geflissentlich übergehen und schnell zum Spiel selbst kommen. Der Spieler steuert ein *Manta*-Raumschiff (Nein, das kann man nicht tieferlegen!), das über die Oberfläche von überdimensionalen Raumschiffen hinwegdüst. Diese sind sehr langgezogen, so daß das Spielfeld horizontal scrollt. Wer jetzt meint, einfach bis zum Ende durchbrettern zu können und dann die Runde geschafft zu haben, täuscht sich. Zuerst muß eine bestimmte Menge gegnerischer Flieger abgeschossen werden. Erst dann darf man sich auf der Landebahn niederlassen und zum nächsten Level vorrücken.

Weltraum-Action einmal anders

Die Geschwindigkeit des *Manta* ist regulierbar. Geübte Piloten legen den Jäger schief oder tauchen mit einer raschen 180-Grad-Wendung einfach unter den Feinden durch. Wenn allerdings eine Mauer im Weg ist, hilft nur Ausweichen. Interessanterweise ist eine Berührung der Feinde harmlos – solange diese nicht schießen. Leider mangelt es den Burschen aber oft an Zurückhaltung, und sie schießen im letzten Moment doch auf das eigene Raumschiff.
Eine flotte Titelmelodie und zahlreiche Soundeffekte sind vorhanden. Leider wird die Farbenpracht des Computers nicht ausgenutzt, die Grafiken sind weitgehend einfarbig. Dennoch sorgen gelungene Schatteneffekte für einen räumlichen Touch. Sehr elegant ist auch, wie das Raumschiff beim Wenden Loopings dreht oder sich schräglegt. Dagegen sehen die Feinde recht langweilig aus.
Das Programm löst sich erfreulicherweise von der Dutzendware durch ein eigenwilliges, dennoch actionlastiges Spielprinzip. Langanhaltender Spielspaß dürfte sicher sein.

URIDIUM 2) 3) 4)

Hersteller: Rack it/Hewson
Steuerung: Joystick oder Tastatur
Voraussetzung: für alle CPCs
Preis: etwa 60 DM
Anleitung: englisch, französisch

Bewertung:

Grafik: 3	Präsentation: 3
Sound: 2	Motivation: 2
Endnote: 1	

Das Szenario von **THUNDER JAWS**: Der Spieler soll in eine Festung unter Wasser eindringen. Denn gemäß eines teuflischen Plans, sollen genetische Mutanten die Welt erobern. Doch der Bösewicht ist diesmal eine Frau – *Madame Q*. Da soll noch einer sagen, den Story-Schreibern würde nichts Neues mehr einfallen.

Die eiskalte Madame Q läßt morden

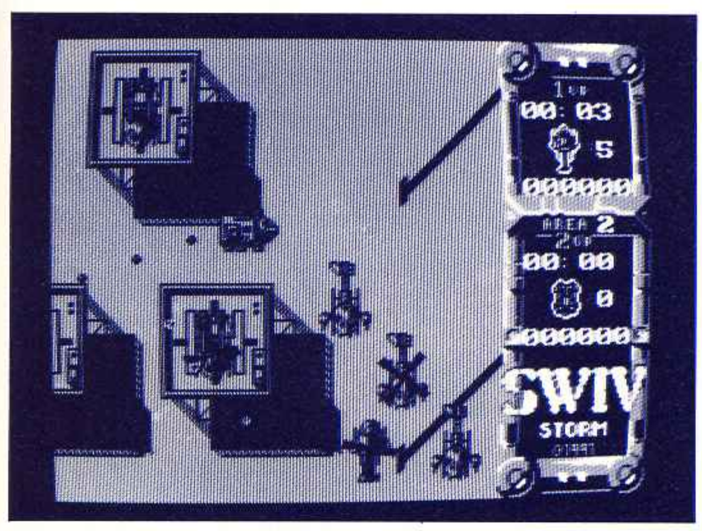
Genug der Vorrede, worum geht es also? Die vier Levels sind in jeweils zwei Abschnitte eingeteilt. Im ersten schwimmen Sie als Froschmann umher und wehren sich gegen Angreifer durch Schußwaffengebrauch. Der Bildschirminhalt scrollt dabei von rechts nach links. Der Ablauf gleicht den etwa 572 anderen Schießspielen dieser Art. THUNDER JAWS ist höchstens etwas langsamer.
Ist man an einer Metalltür angelangt, geht's im Innern der Festung weiter. Wieder schießen, diesmal aber auch springen und ducken. Mit dem als Extra-Waffe herumliegenden Flammenwerfer ist man schlechter bedient als mit dem normalen Schuß.
Was auf dem Meeresgrund und in den Hallen liegt – vom Auto bis zum Überrest eines abgesoffenen Flugzeugs – ist recht bunt, aber häufig doch etwas mickrig ausgefallen.
Ebenso wie an der Originalität wurde an der Schwierigkeit gespart. Die Endgegner jedes Abschnitts sind besonders minderwertig ausgefallen. Wenn man nicht gerade ein Anfänger in Sachen Action-Spiel ist oder was zum "nebenbei mal schnell durchspielen" sucht, sollte man von THUNDER JAWS Abstand nehmen. Das beste sind einige wirklich neue grafische Effekte bei der Ein- und Ausblendung von Bildern.

THUNDER JAWS 2) 3) 4)

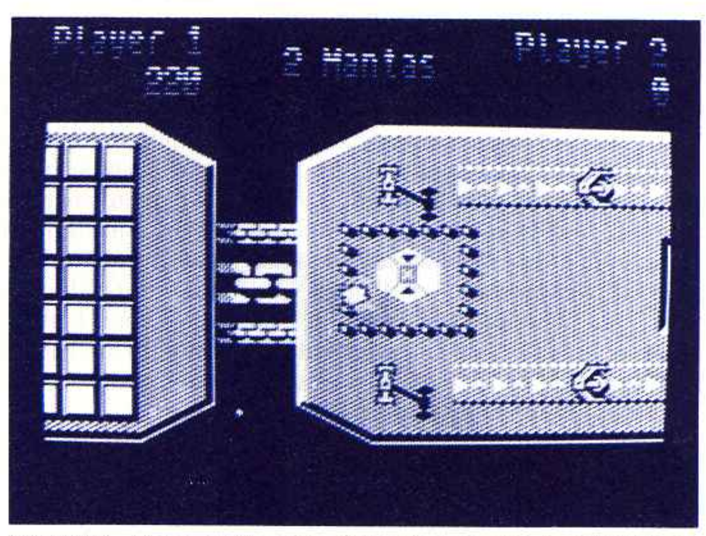
Hersteller: Domark
Steuerung: Joystick oder Tastatur
Voraussetzung: für alle CPCs
Preis: etwa 55 DM
Anleitung: deutsch, französisch
Bemerkung: lädt nach

Bewertung:

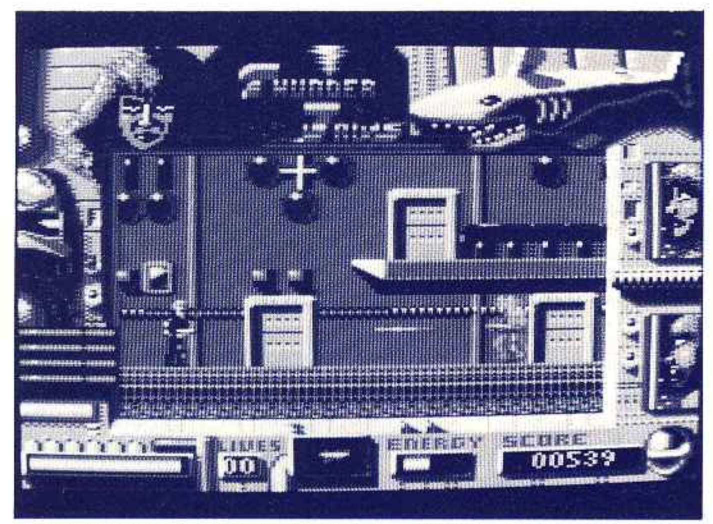
Grafik: 3	Präsentation: 2
Sound: 3	Motivation: 4
Endnote: 4	



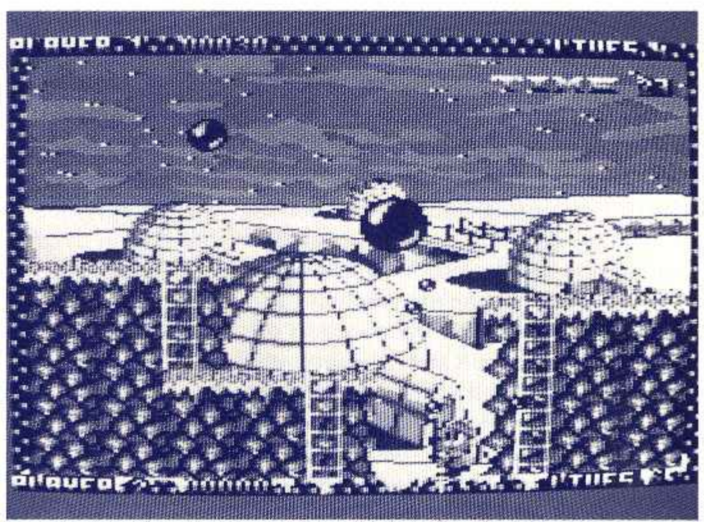
SWIV – Steuern Sie einen Jeep oder einen Hubschrauber im Kampf gegen die feindliche Übermacht (MAX)



URIDIUM – Nur wenn Sie genügend Feinde wegpusten, kommen Sie ans Ziel!



THUNDER JAWS – Lassen Sie sich von Madame Q's Schönheit nicht blenden und durchkreuzen Sie ihre Pläne



ZAP'T'BALLS – THE ADVANCED EDITION – Technische Perfektion und brillantes Spielgeschehen

Wenn sich ein bekannter Demoschreiber daran macht, ein Spiel zu programmieren, darf man etwas Besonderes erwarten. Jetzt kommt **ZAP'T'BALLS – THE ADVANCED EDITION** in die Läden.

Wie schon beim DATABOX-Spiel geht es darum, mit einem *Zapper* Bälle zu eliminieren. Große Bälle verschwinden freilich nicht einfach so mir nichts, dir nichts. Sind sie getroffen, teilen sie sich in zwei kleinere Bälle. Erst die ganz kleinen Baby-Bälle können dann vollständig beseitigt werden.

Kampf dem Fanatismus

Eine leicht schrullige Story liegt dem Spiel zugrunde. Irgendein CPC-plus-Fanatiker kopierte sich selbst in den ASIC-Chip der Plus-Modelle, um von dort aus alle Computer ohne diesen Chip zu vernichten und so den Plus-Rechnern zum Durchbruch zu verhelfen. Dieses Vorgehen ist nicht nur recht unfair, vor allem schadet es auch den alten CPCs. Deshalb machen sich *Ramses* und *Columbus* auf den Weg, dringen mittels eines Cartridges über den Modulschacht in besagten CPC plus ein. Als jetzt der Computer angeschaltet wird, finden sich die beiden Helden in besagtem Spiel wieder und müssen sich darin bewähren.

Neu an der "Advanced Edition" ist, daß es insgesamt 80 Levels in vier Welten mit jeweils unterschiedlichen Hintergrundgrafiken in Top-Qualität gibt.

Hat man die erste Welt geschafft, bekommt man ein Paßwort für Welt 2 und Welt 3. Am Ende dieser beiden Welten gibt es wieder Paßwörter. Nur wenn man beide hat, darf man die letzte der vier Welten betreten. Jede der Welten hat ihre Besonderheiten. In der *World of Ice* gibt es unter anderem Schneeballmaschinen, und bei schlechter Witterung schneit es gar. Auch Glatteis kann sich merklich auf die Steuerung der Spielfigur auswirken.

Die *World of Fire* hat neben vielen anderen Überraschungen bewegliche Blöcke und Sandstürme zu bieten. Diese Extravaganzen in der Kombination mit Leitern und Plattformen sorgen dafür, daß spielerisch jede Menge Abwechslung vorhanden ist. Musikalisch übrigens auch: eine Titelmelodie, die ersten der drei Welten haben eigene Musikstückchen, die auch gegen Sound-Effekte ausgetauscht oder mit diesen kombiniert werden können. Nicht einmal während des Ladens ist der Computer davon abzuhalten, Musik zu machen.

Technisch und spielerisch umwerfend

Doch dies ist nicht das einzige technische Kabinettstückchen. Ein mehrminütiger Vorspann – der sich leider nicht abbrechen läßt – stellt sämtliche gängigen Programmierregeln auf den Kopf. Eine schier unbegrenzte Farbenpracht in Mode 1, Overscan, kombiniert mit Vektorgrafik und Laufschriften. Damit hört es nicht auf. Wie viele Objekte auch auf dem Bildschirm sind, der Spielablauf ist rasant und ohne Ruckeln.

Dank der vielen Besonderheiten der einzelnen Levels, die mal Strategie, mal Geschwindigkeit, mal Reaktion oder einfach eine zündende Idee verlangen, ist für Abwechslung gesorgt.

Der Schwierigkeitsgrad steigert sich, nach jedem zweiten Level gibt's ein Zwischenpaßwort, das es erlaubt, das Spiel künftig dort fortzusetzen. Alles, was bisher gesagt wurde, wäre genug, das Programm mit Lob zu überschütten. Dazu kommt noch, daß die Highscores automatisch auf Diskette gespeichert werden.

In den ersten drei Welten gibt es einen Modus, in dem zwei Spieler gleichzeitig antreten können. Phantastisch – und das nicht etwa bloß, weil wir den Vorgänger herausgebracht haben ...

ZAP'T'BALLS – THE ADVANCED EDITION [1) 3)]

Hersteller: NoName EDV Service
Steuerung: Spieler 1 Joystick, Spieler 2 Tastatur
Voraussetzung: 128 kByte, nur auf Diskette erhältlich
Preis: etwa 50 DM
Anleitung: deutsch, englisch, französisch
Bemerkung: T-Shirt gegen 5,– DM Aufpreis erhältlich
Bewertung:

Grafik: 1	Bildschirmpräsentation: 1 (Verpackung und endgültige Anleitung noch nicht vorliegend)
Sound: 1	Motivation: 1
Endnote: 1	

In einem ganz besonderen Pack wird der **MANIX TWINS**-Joystick zusammen mit der Beau Jolly *BIG BOX* ausgeliefert. Schauen wir uns zuerst einmal den Joystick näher an.

Genaugenommen handelt es sich hier nicht um einen, sondern um zwei miteinander verbundene Sticks. Der Spieler hat ja schließlich zwei Hände. Der eine Griff hat diverse Knöpfe und

Schalter. Außer einem normalen Feuerknopf können hier Dauerfeuer und Schnellfeuer für die einzelnen Buttons eingeschaltet werden. Leider funktioniert das auf dem CPC nicht. In der anderen Hand befindet sich der Griff mit einer Joypad-ähnlichen Konstruktion, die eine Steuerung in die verschiedenen Richtungen zuläßt.
Diese recht bizarre Konstruktion hat während unseres Tests für alles Denkbare gesorgt – nur nicht für gute Ergebnisse. So ist der Joystick nur denen zu empfehlen, die um jeden Preis etwas Originelles wollen, oder für solche, denen Computerspiele zu leicht sind und die so für neue Schwierigkeiten sorgen wollen.
Nun ist da ja noch die BIG BOX. Sie verdient ihren Namen voll und ganz: nicht nur wegen ihrer Abmessungen, die reif fürs Guiness-Buch sind, sondern auch, weil ganze 30 Spiele enthalten sind!
Überraschenderweise ist die Qualität für so eine große Sammlung im Schnitt recht gut. Es finden sich fast alle Spiele, die ACTIVISION und ELCTRIC DREAMS produziert haben.

Manix Twins: Eine Konstruktion für Maniker

Eines der Highlights dürfte zweifellos SPINDIZZY sein. Durch über 350 dreidimensionale Räume wird ein Kreisel (beziehungsweise eine Kugel oder ein Diamant) gesteuert. Damit sollten Juwelen aufgesammelt, Schalter umgelegt und Abgründe vermieden werden. Im Kampf gegen Zeit und trickreiche Konstruktionen geht es um Geschicklichkeit und Einfallsreichtum. Ihrem Forscherdrang sind durch den Umfang kaum Grenzen gesetzt. Inspiriert wurde das Spiel sicherlich von MARBLE MADNESS, doch das Spielprinzip wurde abgewandelt, und das Ergebnis wird von vielen als besser als das Vorbild bezeichnet. Auch nach vielen Jahren noch ein Klassiker und eines der besten CPC-Spiele aller Zeiten.

Besser als das Vorbild

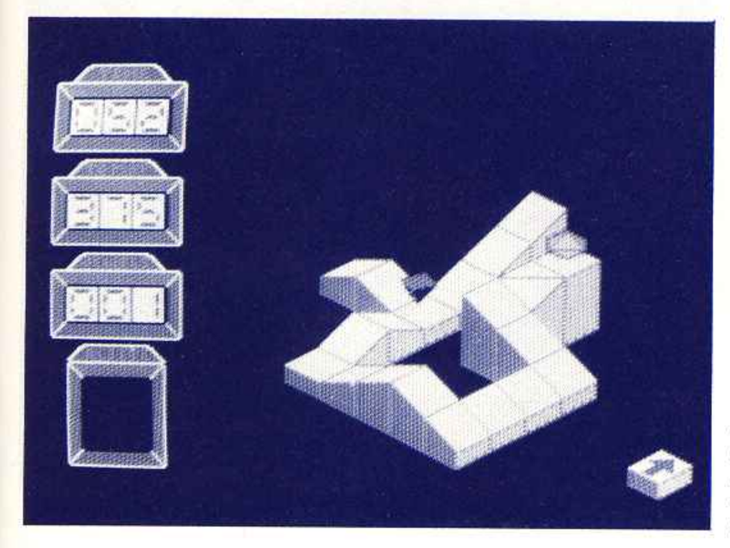
Auch ENDURO RACER, ein Motorradrennen gegen die Zeit, bei dem Sie oft hoch über Schanzen springen müssen, und SUPER SPRINT finden sich in der Sammlung. Bei letzterem sehen Sie die Rennstrecke von oben, können Extras einsammeln. Beide Programme sind nicht mehr taufrisch, brauchen aber den Vergleich mit neueren Programmen kaum zu scheuen.
Technisch nicht gerade perfekt, aber abwechslungsreich sind KARNOV und WONDERBOY, bei denen sich ein Held durch eine scrollende Landschaft kämpft. Um echte Klassiker dürfte es sich bei GHOSTBUSTERS und HACKER II – eines der originellsten Spiele aller Zeiten – handeln.
Für Strategiefans gibt es HIGH FRONTIER (das Spiel zum SDI-Projekt) und GUADALCANAL. Sportfreunde werden mit CHAMPIONSHIP BASEBALL, BASKETBALL und FOOTBALL versorgt. Mit SPACE SHUTTLE liegt sogar eine Simulation bei. Natürlich sind auch ein paar Flops enthalten, aber die Mehrzahl der Programme ist mindestens guter Durchschnitt.

Big Box: Qualität und Quantität

Alle enthaltenen Spiele auf einem Blick: GHOSTBUSTERS, ENDURO RACER, HACKER, BIG TROUBLE IN LITTLE CHINA, WONDERBOY, GFL CHAMPIONSHIP FOOTBALL, GUADALCANAL, ALIENS, SPINDIZZY, RAMPAGE, BARRY McGUIGAN WORLD CHAMPIONSHIP BOXING, EXPLORER, HOWARD THE DUCK, CHAMPIONSHIP BASKETBALL, SUPER SPRINT, KNIGHTMARE, SPACE SHUTTLE, THE REAL GHOSTBUSTERS, GEE BEE AIR RALLY, CHAMPIONSHIP BASEBALL, DANDY, MERMAID MADNESS, STAR RAIDERS II, SAILING, PRODIGY, ISS, HIGH FRONTIER, FIRETRAP, SUPER HANG ON und KARNOV.
Insgesamt gibt es nur einen Grund, warum man sich die BIG BOX nicht zulegen sollte: Bei 30 Spielen auf einmal, wann bleibt da noch Zeit für Schule, Freundin, Arbeit und Familie? Den Joystick kann man ja bei der nächsten sich bietenden Gelegenheit an ungeliebte Verwandte weiterverschenken.

BIG BOX und MANIX TWINS[3) 5)]

Hersteller: Beau Jolly und Dynamix
Voraussetzung: für alle CPCs
Preis: Big Box etwa 55 DM, Manix Twins etwa 35 DM
Anleitung: englisch



SPINDIZZY – Raffinierte 3D-Puzzles sind mit dem Kreisel zu bewältigen (Big Box)

Es gibt zwei Arten von Spielern. Die einen wollen sich zwischendurch nur mal an einem Action-Spektakel abreagieren. Die anderen nehmen sich auch Zeit, um sich in komplexe Programme einzuarbeiten. Wenn Sie zu der zweiten Gruppe gehören, dann dürfte Sie unser Test von **GUNSHIP** interessieren.
Über feindliches Gebiet in einem Hubschrauber zu fliegen, strategisch wichtige Ziele anzuvisieren und zu eliminieren, ist, wie man sich denken kann, nicht ganz einfach. Entsprechend gefordert sind auch die fliegerischen Fähigkeiten des Spielers.
Simuliert wird ein AH-64A Apache Kampfhubschrauber, ein recht modernes Fluggerät. Da sind natürlich die Missionen vom moralischen Gesichtspunkt her etwas kritischer als bei Szenarien aus dem zweiten Weltkrieg. Der Spieler wird überall dorthin geführt, wo sich die Amerikaner einmischen – also rund um den Globus. Besonders schön ist es dann auch zu lesen, daß in einem der Einsätze Deutschland als Schlachtfeld eines Kriegs zwischen NATO und

Warschauer Pakt dient. Glücklicherweise hat die Geschichte hier einen anderen Lauf genommen.
Aber wir wollen hier keine moralischen Abhandlungen führen, sondern sehen, was das Programm bietet. Das ist nämlich eine ganze Menge, da die Simulation nicht oberflächlich, sondern sehr detailliert ist.

Luftkrieg über Deutschland

Drehmoment und Treibstoff, Rotorabkupplung und viele Instrumente zur Navigation, Tag- und Nachtflug sowie leicht lakonische Meldungen bei Mißerfolgen. Alles, was zu einer Hubschraubersimulation gehört, findet man. Um Anfängern den Einstieg zu erleichtern, gibt es einen Modus mit vereinfachter Steuerung.
Fünf verschiedene Missionen sind nicht wahnsinnig viel, aber dank abgestuftem Schwierigkeitsgrad und zahllosen Realitätsniveaus doch für viele Flugstunden genug. Soundmäßig gibt es die obligatorischen Maschinengeräusche und Schüsse sowie Explosionen. Die Landschaft, durch die Sie fliegen, ist grob, einfarbig und ohne Schattierungen. Schön und ausführlich ist jedoch das Handbuch, das auch eine Menge Hintergrundinformationen liefert. Auch eine Tastaturschablone liegt bei. Sie wird auf die Tastatur gelegt, so daß neben jeder Taste eine Beschreibung der Funktion steht.
Simulator-Freunde mit einem CPC haben sowieso keine übergroße Auswahl. Wenn es etwas anderes als ein Flugzeug ist, das simuliert werden soll, lichten sich die Reihen noch mehr. Um so erfreulicher, daß GUNSHIP spielerisch fast alle Wünsche befriedigt.

GUNSHIP 1) 2) 3) 4)

Hersteller: Microprose
Steuerung: Tastatur oder Joystick und Tastatur
Voraussetzung: für alle CPCs
Preis: etwa 70 DM
Bemerkung: Texte auf dem Bildschirm französisch, Tastaturschablone englisch; lädt nach.
Anleitung: deutsch

Bewertung:

Grafik: 4	Präsentation: 2
Sound: 4	Motivation: 2
Endnote: 2	

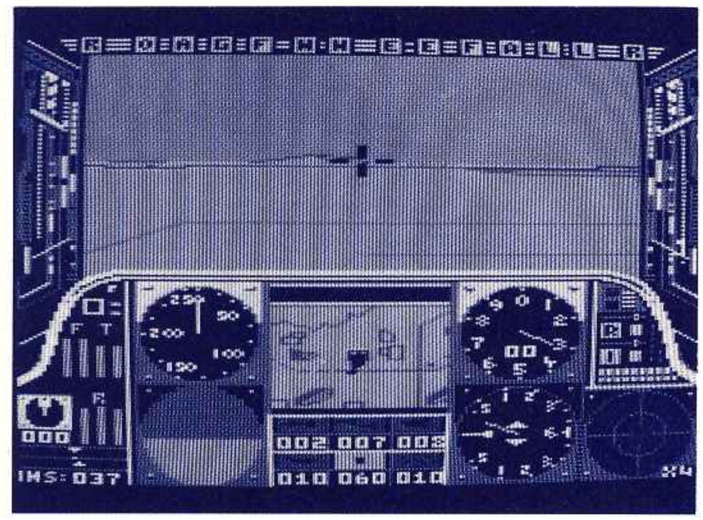
Eines der ersten ernstzunehmenden Kampfsportspiele war THE WAY OF THE EXPLODING FIST. Das jetzt wiederveröffentlichte **INTERNATIONAL KARATE+** ist eigentlich eine Weiterentwicklung davon. Die Spielerfiguren sehen denen von EXPLODING FIST zum Verwechseln ähnlich. Inzwischen haben sie aber noch einige neue Kampftechniken und Bewegungen gelernt, darunter den eleganten Rückwärts-Flic-Flac.

Drei Karatekas kämpfen um die höchsten Auszeichnungen

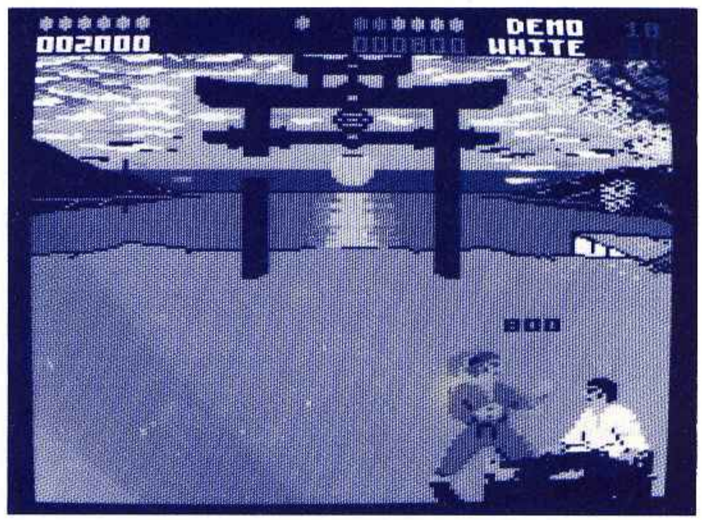
Clou der Sache ist aber, daß sich nicht zwei, sondern drei Kämpfer auf dem Bildschirm tummeln. Der weiße Kämpfer wird vom ersten Spieler gesteuert, der blaue vom Computer. Wenn ein zweiter Spieler teilnimmt, steuert dieser den roten Karateka. Spielen Sie allein, übernimmt der Computer diese Aufgabe. Im Kampf heißt es dann jeder gegen jeden. Denn wer nach einer Kampfrunde das schlechteste Ergebnis hat, scheidet aus und wird von einem Computerkämpfer ersetzt. Natürlich werden sich mehrere menschliche Spieler dann doch absprechen, zunächst den Computer flachzulegen.
Ebenfalls vom Vorbild EXPLODING FIST übernommen wurde ein Schiedsrichter, der den Gewinner nennt, außerdem das Bewertungssystem, das schwierige Schläge höher bewertet als einfache. In den höheren Levels ändert sich nicht viel, aber die Computergegner werden geschickter, so daß man immer raffinierter vorgehen muß, um sie zu besiegen.

Raffiniertes Vorgehen ist gefragt

Zwischen den Kämpfen gibt es Bonusrunden, in denen umherhüpfende Bälle abgewehrt werden sollen, um Bonuspunkte zu scheffeln.
Das Hintergrundbild zeigt einen Sonnenuntergang, der sich in einem Gewässer spiegelt. Die Spiegelung der Sonne ist jedoch nicht starr, sondern bewegt sich, um Wellen anzudeuten. Dazu gibt es Kampfgeräusche, und eine Musik läuft im Hintergrund, die mit vielen Effekten aufwartet (wobei allerdings die eigentliche Melodie etwas zu kurzkommt).
Trotz der drei Kämpfer ist der Ablauf flott, und die 17 verschiedenen Schlagtechniken stellen sämtliche aktuellen Prügelspiele, die oft mit drei Schlägen auskommen, in den Schatten. Wer in EXPOLDING FIST geübt ist, hat wenig Schwierigkeiten, sich bis zu höchsten Auszeichnungen hochzukämpfen.



GUNSHIP – Eine erstklassige Hubschraubersimulation.



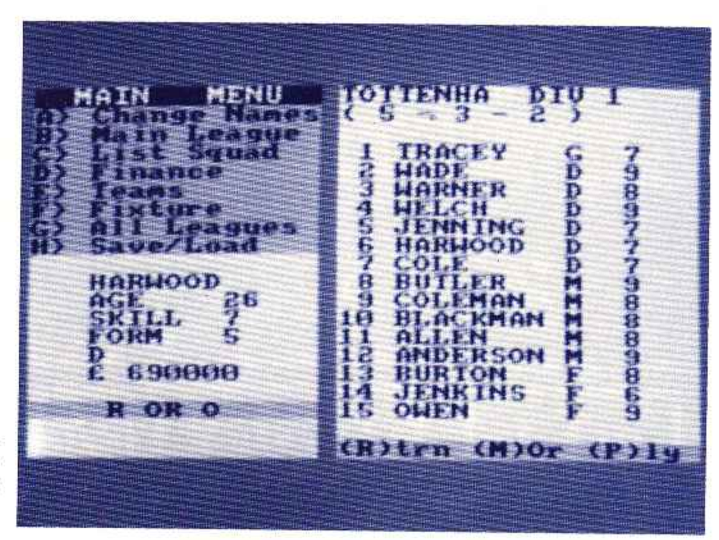
INTERNATIONAL KARATE – Holen Sie sich den höchsten Dan-Rang!

Alle anderen bekommen mit IK+ das wohl beste Karatespiel für die kleinen Amstrads in die Hand.



INTERNATIONAL KARATE 1) 2) 3) 4)

Hersteller: Hit Squad
Steuerung: Joystick (Spieler 1), Tastatur (Spieler 2)
Voraussetzung: für alle CPCs
Preis: etwa 60 DM
Anleitung: deutsch, englisch
Bewertung:
Grafik: 2 Präsentation: 3
Sound: 2 Motivation: 2
Endnote: 2



SOCCER 6 – Ein halbes Dutzend Managementspiele auf einen Streich

Auch wenn's für Deutschland bei der Europameisterschaft nicht ganz zum Titel gereicht hat, kommen wir nun zu **SOCCER 6**. Hier sind auf einer Kassette gleich sechs verschiedene Fußballmanagement-Spiele zusammengefaßt. Sinn der Sache ist, mit dem eigenen Verein möglichst viel Erfolg zu haben. Abgesehen von der sportlichen Seite rund ums Training, dreht sich auch alles ums liebe Geld.
Je nachdem, welches Spiel man spielt, liegt der Schwerpunkt unterschiedlich. Training im einen, der Transfermarkt im anderen und taktische Erwägungen bei der Mannschaftsaufstellung im dritten. Ist ja alles schön und gut, aber ein Programm statt sechsen, dafür mit allen Funktionen vereint, hätte zwar zu weniger Masse, dafür aber mehr Klasse geführt.

Gespielt wird 90 Minuten ...

Eine kleine Kuriosität ist die Meldung während des Ladens, man solle doch bitte CAPS LOCK drücken. Schönen Gruß an die Programmierer: Der Befehl UPPER$ – die Programme sind sämtlich in BASIC, was aber nicht weiter stört – wandelt alle Eingaben in Großbuchstaben um. Außerdem gibt's einen POKE, der CAPS LOCK einschaltet. Warum also nicht eine dieser Möglichkeiten verwenden?

... und der Ball ist rund!

Daß auf dem Bildschirm keine Grafik, sondern nur Text zu sehen und keinerlei Sound vorhanden ist, läßt sich bei Management-Spielen relativ leicht verschmerzen. Von einer eleganten Benutzerführung kann man kaum sprechen. Hier gibt es also ebenfalls keine Pluspunkte zu vergeben.
Wer es nicht allzu komplex mag und sich an Managementspielen erfreut, bekommt mit SOCCER 6 eine ganze Menge Futter. Die einzelnen Spiele unterscheiden sich – soweit dies innerhalb dieses eng begrenzten Genres möglich ist – relativ stark. Leider ist aber keines der enthaltenen Programme in irgendeiner Form herausragend. So bleibt das positivste Merkmal der annehmbare Preis.

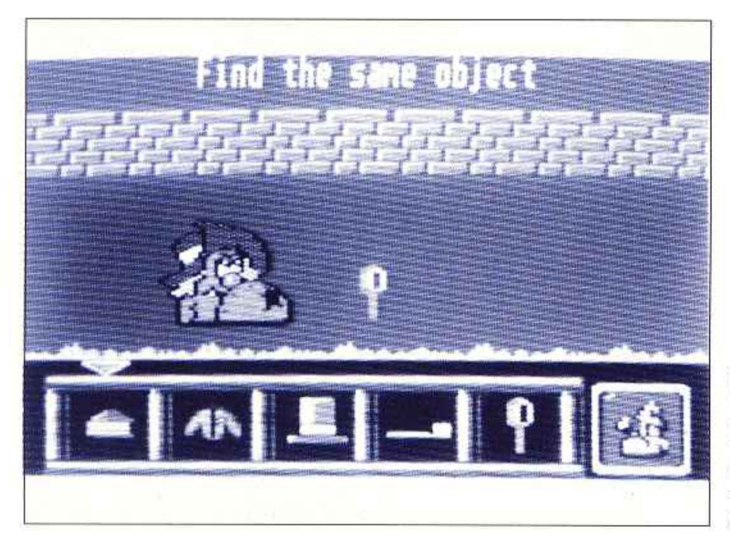
SOCCER 6 1) 3) 4)

Hersteller: Challenge
Steuerung: Tastatur
Voraussetzung: für alle CPCs
Preis: etwa 60 DM
Anleitung: englisch
Bewertung:
Grafik: - Präsentation: 5
Sound: - Motivation: 4
Endnote: 3

Der Gang ist kalt. Je weiter *Tanis* geht, desto düsterer wird es. Dann werden seine Befürchtungen wahr. Schwere Schritte nahen heran. Der erschöpfte *Tanis* wird vom selben Schicksal ereilt wie seine sieben Mitstreiter aus **HEROS OF THE LANCE** – im Kampf gefallen, in einem verlassenen Ende eines Verlieses, wo ihn nie jemand finden wird.
Wer soll nun die *Disks of Mishkal* in den Ruinen von *Xak Tsaroth* finden? Ohne sie wird es kaum möglich sein, den Glauben an die alten Götter wiederherzustellen. Dann wird die Königin der Nacht aus dem Abgrund heraufsteigen und die Welt in ewige Dunkelheit hüllen, das teuflischste Reich aufbauen, das es je gab. Schon ihre niedersten Kreaturen, die sie vorgeschickt hat, lassen wenig Gutes hoffen.
Spätestens jetzt ist es wohl klar geworden, daß es sich bei HEROS OF THE LANCE um ein Fantasy-Spiel handelt. Als Vorlage dient die Advanced Dungeons & Dragons (AD & D) Serie.
Die klassischen Computerrollenspiele folgen meist zwei Schemata. Entweder wird die ganze Welt dreidimensional aus der Sicht des Spielers dargestellt – der bekannteste Vertreter dieser Art ist BARD'S TALE. Oder, wie die ULTIMA-Serie, in Draufsicht. Häufig gibt es Adventure-Anleihen, das Ergebnis der Kämpfe wird durch den Zufall und die Charaktereigenschaften bestimmt.
Von diesen Formen löst sich nun HEROS OF THE LANCE. Im Arcade-Adventure-Stil werden die Handlungen des Spielers von der Seite gezeigt. Alle Bewegungen, wie Laufen und Springen sowie Nahkampf und Angriffe aus größeren Entfernungen werden direkt mit dem Joystick ausgeführt. Das wird dann auch gleich in detaillierter, flüssig animierter Mode-1-Grafik gezeigt. Für speziellere Aktionen wie den Einsatz von Magie oder das Einsammeln von herumliegenden Objekten wird mit der Leertaste ein Menü auf den Bildschirm geholt.

Fantasy-Rollenspiel einmal anders

Ein Kompaß zeigt an, wo sich welche Himmelsrichtung befindet. Das ändert sich nämlich, wenn man nicht nur links und rechts, sondern auch hoch und runter geht. Das zu beschreiben, ist recht aufwenig, wenn man spielt, merkt man aber bald, wie es funktioniert. Trotzdem ist es unerläßlich, eine Karte zu



FIRST CLASS WITH THE SHOE PEOPLE – Daß man am Computer auch etwas lernen kann, beweist dieses Spiel für die ganz Kleinen

zeichnen. Sonst hat man sich bald hoffnungslos verirrt.
Um Erfolg zu haben, muß man die acht Figuren geschickt einsetzen. Für Faustkämpfe sollte man einen kampferprobten Haudegen vorschicken. Dieser erhält dann vielleicht magische Unterstützung von einem körperlich schwachen Zauberer aus der hinteren Reihe.
Das Suchen von Wegen und Gegenständen ist ein wichtiges Element des Spiels. Hinzu kommen Kämpfe, Magie, Zaubertränke und Ringe, aber auch Anleihen aus dem Action-Sektor wie von der Decke fallende Steine. Weil die Aufgabe recht umfangreich ist, kann der Spielstand jederzeit gespeichert werden.
Eine interessante Mischung und eines der gehaltvollsten Spiele, die jetzt als Billigspiel wiederveröffentlicht wurden. Wer die Fantasy-Spiel-Atmosphäre mag, bekommt mit HEROS OF THE LANCE ein Produkt, an dem er lange und mit Begeisterung zu knabbern hat.

HEROS OF THE LANCE[1)3)4)5)]

Hersteller: Kixx
Steuerung: Joystick und Tastatur
Voraussetzung: für alle CPCs
Preis: etwa 60 DM
Anleitung: englisch, italienisch
Bemerkung: lädt nach

Bewertung:
Grafik: 3 Präsentation: 2
Sound: 3 Motivation: 1
Endnote: 1

"Computerspiele sind hirnlos und führen zur Verblödung" lautet ein gängiges Vorurteil. Schon fast so alt wie Homecomputer sind auch elektronische Vokabeltrainer, doch **THE SHOE PEOPLE** von Gremlin Graphics wendet sich an die ganz Jungen. Als Alter der Zielgruppe ist 4 bis 6 Jahre angegeben.
Die Kinder sollen Gegenstände unterscheiden und deren Namen lernen, Groß- und Kleinbuchstaben unterscheiden und in niedrigen Zahlenbereichen addieren und subtrahieren lernen. Insgesamt sechs Teilprogramme gibt es, die alle verschiedene Fähigkeiten trainieren sollen und jeweils noch einstellbare Schwierigkeitsstufen aufweisen.
Ein Extra-Bonbon zur Förderung der Kreativität soll das elektronische Malbuch sein. Damit die ganze Sache Spaß macht, ist das Programm grafisch kindgerecht und süß aufbereitet worden. Wenn eine Aufgabenstellung besonders erfolgreich gelöst wurde, gibt's als Belohnung eine besondere grafische Sequenz. Auch die Bedienung dürfte sogar den jüngsten Kids kaum Schwierigkeiten bereiten: ENTER und die LEERTASTE reichen meist aus.

Papa, der Computer hat gesagt ...

Ein kleines Heftchen und ein Anstecker mit den Hauptfiguren des Spiels liegen bei. Sie sind die Helden in englischen Kinderheftchen. Bei dem Wort "englisch" sind wir aber auch gleich beim größten Haken des Spiels. Anleitung und Bildschirmtexte sind allesamt in Englisch gehalten. In unseren Breiten dürfte aber kaum ein Kind im entsprechenden Alter des Englischen mächtig sein.
Viele der Bildschirmanweisungen sind nur Einstellungen vor Spielbeginn, können also auch von einem Erwachsenen vorgenommen werden. Doch bei vielen Übungen geht es auch um das Auffinden von passenden Wörtern. Diese Programmteile sind wohl nur dann brauchbar, wenn es darum geht, einen jungen Englischanfänger behutsam an die neue Sprache heranzuführen.
Die anderen Übungen sind aber sicher für die Kleinen dank der hübschen Aufbereitung interessant, so daß das Lernen quasi spielerisch und nebenbei erfolgt. Doch Achtung, auch kleine Kinder sind oft schneller vom Computer und Spielen fasziniert, als den Eltern lieb ist, und auch dann nicht vom Autorennen oder Fußballspiel wegzubringen, wenn der Papi gerade seine Textverarbeitung braucht!

FIRST CLASS WITH THE SHOE PEOPLE[3)]

Hersteller: Gremlin Graphics
Steuerung: Tastatur
Voraussetzung: mindestens ein Kind im Alter von 4 bis 6 Jahren, irgendein CPC
Preis: etwa 60 DM
Anleitung: englisch

Andreas Lober/jg

Die Muster stellten uns zur Verfügung:

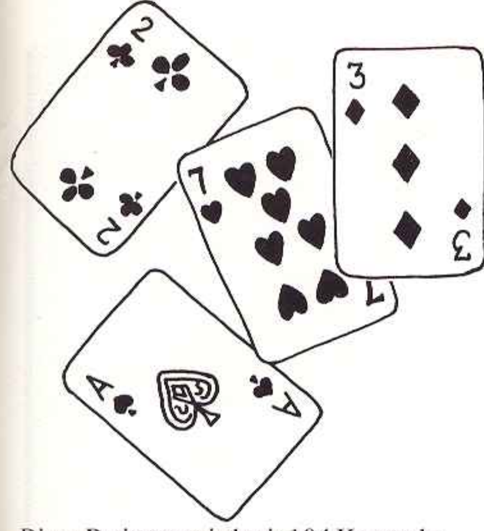
[1)] NoName EDV Service
Weidenweg 14
2905 Edewecht

[2)] Weeske Computer-Elektronik
Potsdamer Ring 10
W-7150 Backnang

[3)] EDV Obermeier
Bünder Straße 20
4972 Löhne 1

[4)] Brostedt
Computertechnik
Harkortstr. 54
4690 Herne 2

[5)] LYNX Versand
Schevenstr. 24
4650 Gelsenkirchen 1



Die Schwierige

Patiencen legen mit Komfort

Die Schwierige ist eine der schönsten und – wie der Name bereits sagt – schwierigsten Patiencen. Auch Sie können jetzt in den Genuß einer gelungenen Computerumsetzung kommen.

Diese Patience wird mit 104 Karten bestehend aus zwei Kartdendecks zu je 52 Karten gelegt. Ihre Aufgabe besteht darin, auf den acht Assen aufsteigend über 2,3 bis X, B, D und K die einzelnen Farben zu sammeln. In der Mitte bleibt der Platz für die acht Asse frei. Hierbei gilt, daß jeweils zwei gleiche nebeneinander liegen dürfen.
Oben beginnend mit Kreuz, darunter Pik, Herz und Karo.

Von As bis König

Zu Anfang werden acht Karten vom gut gemischten Kartenblock links und rechts in vier Reihen an die Plätze, die für die Asse freigehalten sind, angelegt. Jetzt kann der Spieler entscheiden, welchen Weg er wählt. Er kann Asse und später auf diese passende Folgekarten sofort auf die acht mittleren Stapel legen. Das geschieht, indem er erst die Ziffer der Reihe eingibt, aus welcher die Karte genommen werden soll. Darauf folgt der Buchstabe des Stapels, auf den er die Karte legen will. Es kann auch eine Karte in eine andere Reihe gelegt werden; vorausgesetzt, diese Reihe ist frei, oder die Karte paßt an die letzte dort bereits liegende.
Eine Karte paßt, wenn sie in absteigender Ordnung angelegt werden kann. Also die Vier an die Fünf, Zehn (X) an den Buben (B) und so weiter. Zusätzlich muß sie aber auch noch die Farbe wechseln. An eine rote Karte (Herz oder Karo) kann folglich nur eine schwarze (Pik oder Kreuz) angelegt werden.

Karten in den Keller

Kann man einmal nichts mit einer Karte anfangen, kann sie in den Keller (9+0) geschoben und dort auf Halde gehalten werden. Bei Bedarf nimmt man sie von dort einfach wieder heraus und legt sie in die entsprechende Reihe.
Es ist auch erlaubt, Karten im Block von einer Reihe in die andere zu bringen.
Ein Block ist hier eine Folge von mindestens zwei aneinander passenden Karten. Dazu gibt man erst die Ziffer der Reihe ein, darauf das Zeichen "+". Nun kann mit "+" oder "-" in der Reihe hin- und hergefahren werden. Hiermit sucht man die Karte aus, bis zu welcher der Block umgelegt werden soll.
Durch Druck auf <ENTER> oder <RETURN> braucht nur noch die Reihe eingegeben werden, in die der Kartenblock kopiert werden soll. Es versteht sich von selbst, daß diese Operation nur von Reihe zu Reihe möglich ist. Bedingung ist auch hier, daß die innerste Karte des Blocks an die äußere Karte der empfangenden Reihe paßt.

Auf ein Neues

Sind die ersten acht ausgelegten Karten genügend hin- und hergeschoben worden, können mit <ENTER> oder <RETURN> die nächsten acht Karten angefordert werden. Das geht so lange, bis alle 104 Karten ausgelegt sind. Dann kann durch geschicktes Umlegen weiter versucht werden, schließlich alle Karten von AS bis König auf den mittleren acht Stapeln zu versammeln. Damit ist das Spielziel erreicht.
In der linken oberen Ecke wird immer angezeigt, wie viele Möglichkeiten zur Verfügung stehen.

Günter Kramer/rs

Übrigens: Das Programm PATIENCE.BAS muß immer vorgeladen werden. Um Ihnen die Vielfalt von Patiencen auf Ihren Monitor bringen zu können, finden Sie auf der DATABOX zwei weitere Varianten mit den Namen: "Zickzack-Kurs" und "Der Zopf".

```
<18> 100 ' Patience.Bas
<20> 110 ' Joyce PCW 8256/8512/9512
<22> 120 ' (c) 1992 Günter Kramer
<50> 130 '....... & CPC International
<24> 140 GOSUB 1640
<16> 150 PRINT cls$;cur.aus$;
<66> 160 GOSUB 1760
< 8> 170 GOSUB 2070
<12> 180 GOSUB 260
< 0> 190 GOSUB 430
<39> 200 GOSUB 840
<10> 210 PRINT cur.ein$;cls$
<28> 220 END
<46> 230 ******************************
<57> 240 *** Variable + Dimensionen ***
<50> 250 ******************************
<61> 260 DIM legen(12,52)
<21> 270 DIM ablage(10)
<55> 280 DIM folge(103)
<64> 290 DIM erstk(8)
<76> 300 DIM pro$(3,2)
< 5> 310 RESTORE 2300
<30> 320 FOR i=1 TO 3
```

```
<53> 330 :. READ pro$(i,0),pro$(i,1)
<71> 340 :. pro$(i,2)="0"
<46> 350 NEXT
<30> 360 FOR i=1 TO 10
<50> 370 :. ablage(i)=99
<52> 380 NEXT
<87> 390 RETURN
< 0> 400 *****************
<70> 410 *** Titelbild ***
< 4> 420 *****************
<86> 430 PRINT cls$;
<71> 440 oben$=CHR$(150)+STRING$(5,154)+CHR$(156)
<29> 450 oben$=oben$+SPACE$(43)+oben$
<19> 460 unten$=CHR$(147)+STRING$(5,154)+CHR$(153)
<79> 470 unten$=unten$+SPACE$(43)+unten$
<76> 480 reihe$=CHR$(149)+SPACE$(5)+CHR$(149)
<79> 490 rechts$=RIGHT$(reihe$,6)
<21> 500 links$=LEFT$(reihe$,6)
<59> 510 as$=CHR$(149)+SPACE$(2)+CHR$(160)+SPACE$(
     2)+CHR$(149)
<42> 520 as$=as$+SPACE$(17)+"Patiencen"+SPACE$(17)
     +as$
<37> 530 zeile=5:spalte=20
```

```
<39> 540 a$=oben$
< 2> 550 GOSUB 790
<15> 560 zeile=6
<28> 570 a$="A"+rechts$+SPACE$(43)+"A"+rechts$
< 8> 580 GOSUB 790
< 2> 590 a$=CHR$(160)+rechts$+SPACE$(43)+CHR$(161)
     +rechts$
<15> 600 zeile=7
<94> 610 GOSUB 790
<62> 620 a$=as$
<32> 630 zeile=8
< 1> 640 GOSUB 790
<96> 650 a$=CHR$(161)
<45> 660 spalte=73
< 7> 670 GOSUB 790
<23> 680 a$=links$+CHR$(160)+SPACE$(43)+links$+CHR
     $(161)
<94> 690 zeile=9:spalte=20
<93> 700 GOSUB 790
<68> 710 PRINT FNcursor$(10,20)links$"A"SPACE$(43)
     links$"A"
<27> 720 PRINT FNcursor$(11,20)unten$
<85> 730 PRINT FNcursor$(12,40)"programmiert von"
<90> 740 PRINT FNcursor$(14,41)"Günter. Kramer"
<83> 750 RETURN
<24> 760 *******************
< 4> 770 *** Bildausgabe ***
<28> 780 *******************
< 3> 790 PRINT FNcursor$(zeile,spalte)a$
<74> 800 RETURN
<15> 810 ****************
<56> 820 *** Abfragen ***
<19> 830 ****************
<84> 840 a$=CHR$(134)+STRING$(21,138)+CHR$(140)
<83> 850 zeile=16:spalte=36
< 7> 860 GOSUB 790
<13> 870 a$=CHR$(133)+SPACE$(21)+CHR$(133)
<95> 880 FOR zeile=17 TO 24
<56> 890 :. GOSUB 790
<42> 900 NEXT
<83> 910 a$=CHR$(131)+STRING$(21,138)+CHR$(137)
< 9> 920 zeile=25
< 2> 930 GOSUB 790
<51> 940 zeile=18:spalte=38
<53> 950 a$="Bitte eingeben:"
< 8> 960 GOSUB 790
<58> 970 zeile=20
<48> 980 FOR i=1 TO 3
< 2> 990 :. a$=pro$(i,1)
<56> 1000 :. b$=FIND$(pro$(i,0))
< 1> 1010 :. IF b$<>"" THEN GOSUB 790:zeile=zeile+
     1:pro$(i,2)="1"
<88> 1020 NEXT
<12> 1030 a$="0 = Ende"
< 8> 1040 GOSUB 790
<79> 1050 a$="1"
<37> 1060 WHILE a$="1"
< 8> 1070 :. b$=INKEY$
< 8> 1080 :. IF b$="0" THEN PRINT cur.ein$:END
<41> 1090 :. a=0
<48> 1100 :. IF b$<>"" THEN a=VAL(b$)
<83> 1110 :. IF pro$(a,2)="1" THEN a$=""
<19> 1120 WEND
<38> 1130 FOR i=0 TO 51
<95> 1140 :. folge(i)=i
<40> 1150 :. folge(i+52)=i
< 3> 1160 NEXT
<53> 1170 ON a GOSUB 1260,1430,1550
< 1> 1180 a$="2"
<16> 1190 WHILE a$<>""
<81> 1200 :. a$=INKEY$
<18> 1210 WEND
<51> 1220 CHAIN MERGE pro$(a,0),0,ALL,DELETE
<15> 1230 **********************************
<41> 1240 *** vorbereiten die Schwierige ***
<21> 1250 **********************************
<82> 1260 b=0
<25> 1270 GOSUB 1320
< 7> 1280 RETURN
<96> 1290 ************************
<72> 1300 *** sortieren Karten ***
<74> 1310 ************************
<37> 1320 FOR i=b TO 103
<80> 1330 :. c=i
<86> 1340 :. WHILE i=c
<29> 1350 :. :. c=b+(INT(RND(1)*(104-b)))
<85> 1360 :. WEND
<87> 1370 :. SWAP folge(i),folge(c)
<13> 1380 NEXT
<12> 1390 RETURN
<55> 1400 *********************************
<82> 1410 *** vorbereiten Zickzack-Kurs ***
<61> 1420 *********************************
<42> 1430 b=8
<34> 1440 FOR i=1 TO 3
<39> 1450 :. SWAP folge(i),folge(i*13)
< 9> 1460 NEXT
<48> 1470 FOR i=4 TO 7
<35> 1480 :. SWAP folge(i),folge((i-3)*13-1)
<18> 1490 NEXT
<10> 1500 GOSUB 1320
<91> 1510 RETURN
<81> 1520 ************************
<55> 1530 *** vorbereiten Zopf ***
<87> 1540 ************************
<50> 1550 b=8
<11> 1560 FOR i=1 TO 7
<47> 1570 :. SWAP folge(i),folge(i*13)
<17> 1580 NEXT
<37> 1590 GOSUB 1320
<90> 1600 RETURN
<50> 1610 ********************************
<80> 1620 *** Bildschirm-Steuerzeichen ***
<56> 1630 ********************************
<47> 1640 esc$=CHR$(27)
<50> 1650 home$=esc$+"H"
<65> 1660 cls$=esc$+"E"+home$
<40> 1670 invers$=esc$+"p"
<91> 1680 normal$=esc$+"q"
<62> 1690 cur.ein$=esc$+"e"
<12> 1700 cur.aus$=esc$+"f"
<94> 1710 DEF FNcursor$(zeile,spalte)=esc$+"Y"+CHR
     $(32+zeile)+CHR$(32+spalte)
<98> 1720 RETURN
<76> 1730 **************************
<27> 1740 *** Zeichensatz ändern ***
<82> 1750 **************************
<16> 1760 PRINT FNcursor$(15,40)"Bitte etwas Gedul
     d"
<19> 1770 zeisa=HIMEM-30
<31> 1780 MEMORY zeisa
<23> 1790 zeisa=zeisa+1
<13> 1800 RESTORE 2200
<40> 1810 FOR i=zeisa TO zeisa+20
<53> 1820 READ code$
<91> 1830 POKE i,VAL("&H"+code$)
<11> 1840 NEXT
<34> 1850 code=zeisa+9
<71> 1860 POKE zeisa+1,code-INT(code/256)*256
<77> 1870 POKE zeisa+2,INT(code/256)
<14> 1880 code=zeisa+21
< 0> 1890 POKE zeisa+10,code-INT(code/256)*256
<17> 1900 POKE zeisa+11,INT(code/256)
<66> 1910 RESTORE 2230
<22> 1920 FOR i=160 TO 163
< 8> 1930 FOR j=0 TO 7
<63> 1940 READ zeich
<76> 1950 POKE code+j,zeich
<44> 1960 a=i*8+&HB800+2^16
<68> 1970 POKE zeisa+16,a-INT(a/256)*256
<75> 1980 POKE zeisa+17,INT(a/256)
<28> 1990 NEXT
<96> 2000 CALL zeisa
<86> 2010 NEXT
<66> 2020 MEMORY HIMEM+30
<88> 2030 RETURN
<24> 2040 ***************************
<96> 2050 *** Uhrzeit + RANDOMIZE ***
<30> 2060 ***************************
<87> 2070 stunde$=HEX$(PEEK(&HFBF6))
<74> 2080 minute$=HEX$(PEEK(&HFBF7))
<44> 2090 sekunde$=HEX$(PEEK(&HFBF8))
<78> 2100 stunde=VAL(stunde$)
<79> 2110 minute=VAL(minute$)
<34> 2120 sekunde=VAL(sekunde$)
<33> 2130 RANDOMIZE (stunde+minute+sekunde)/(sekun
     de+1)
<93> 2140 RETURN
<41> 2150 :. folge(i+52)=i
< 4> 2160 NEXT
<59> 2170 ********************************
<53> 2180 *** Datas Zeichensatz ändern ***
<65> 2190 ********************************
<86> 2200 DATA 01,ff,ff,cd,5a,fc,e9,00,c9
< 3> 2210 DATA 21,ff,ff,01,08,00,11,ff,ff
<95> 2220 DATA ed,b0,c9
<76> 2230 DATA 0,16,56,68,108,68,16,56
<43> 2240 DATA 0,16,56,124,254,108,16,56
```

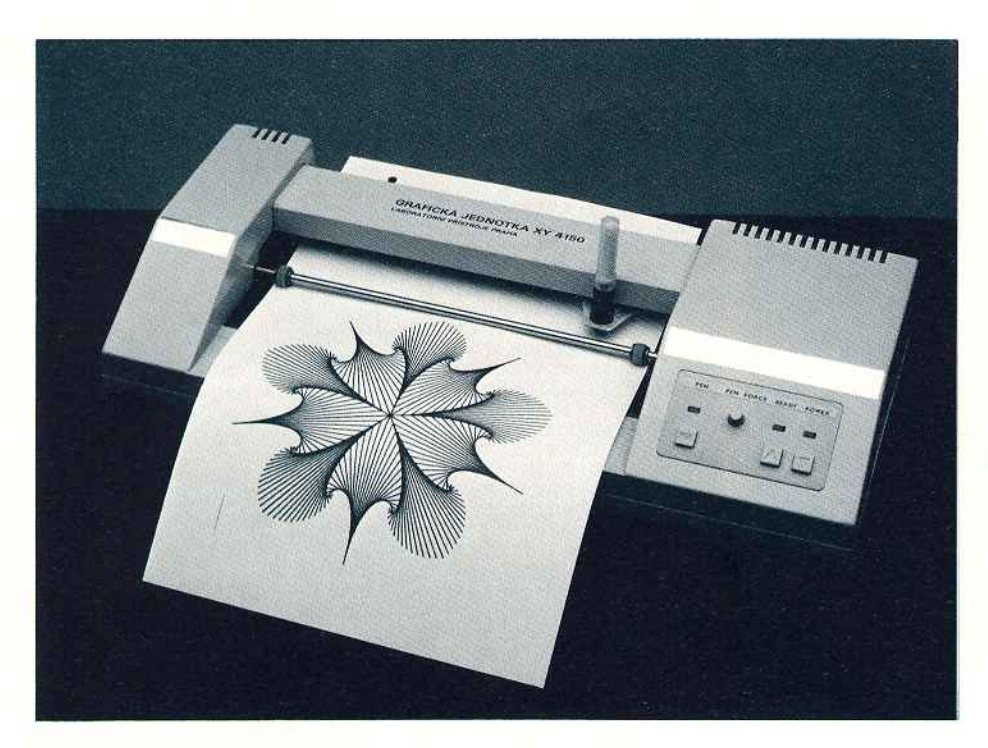
```
<74> 2250 DATA 0,108,254,254,124,56,16,0
<76> 2260 DATA 0,16,56,124,254,254,124,56,16
<28> 2270 *********************
<95> 2280 *** weitere Datas ***
<34> 2290 *********************
<24> 2300 DATA "schwieri.bas","1 = Die Schwierige"
< 9> 2310 DATA "zickzack.bas","2 = Zick-zack-Kurs"
<47> 2320 DATA "zopf.bas","3 = Der Zopf"
```

```
<18> 100 ' Schwieri.Bas
<20> 110 ' Joyce PCW 8256/8512/9512
<22> 120 ' (c) 1992 Günter Kramer
<50> 130 '....... & CPC International
<87> 140 GOSUB 610
< 6> 150 GOSUB 260
<52> 160 GOSUB 660
<95> 170 GOSUB 1160
<60> 180 a$=""
< 7> 190 WHILE a$=""
<94> 200 :. a$=INKEY$
<82> 210 WEND
<69> 220 RUN "patience.bas"
< 4> 230 *****************
<65> 240 *** titelbild ***
< 8> 250 *****************
<96> 260 PRINT cls$
<72> 270 FOR k=0 TO 12
<16> 280 :. FOR i=0 TO 3
<90> 290 :. :. zeile=8*i
<76> 300 :. :. karte=i*13+k
<39> 310 :. :. spalte=k*6+5
<81> 320 :. :. GOSUB 1640
< 7> 330 :. NEXT
<44> 340 NEXT
<86> 350 a$="invers"
<29> 360 IF invers=0 THEN a$="normal"
<27> 370 PRINT "F1 = Karten "a$" ausgeben - - ande
     re Taste = weiter"
<84> 380 zeile=6:spalte=17
< 1> 390 a$="Die Schwierige"
<29> 400 GOSUB 1920
<58> 410 a$="Programmiert von"
<49> 420 zeile=14:spalte=36
<35> 430 GOSUB 1920
< 5> 440 zeile=22:spalte=57
<27> 450 a$="Günter. Kramer"
<41> 460 GOSUB 1920
<38> 470 WHILE a$<>""
<13> 480 :. a$=INKEY$
< 1> 490 WEND
<92> 500 WHILE a$=""
< 0> 510 :. a$=INKEY$
<65> 520 :. IF a$<>CHR$(26) THEN 560
<46> 530 :. a$=""
<61> 540 :. IF invers=0 THEN invers$=esc$+"q":inve
     rs=1:GOTO 270
<76> 550 :. invers=0:invers$=esc$+"p":GOTO 270
<95> 560 WEND
<85> 570 RETURN
<44> 580 ********************
<66> 590 *** Definitionen ***
<29> 600 ********************
<53> 610 DEF FNcursor$(zeile,spalte)=esc$+"Y"+CHR$
     (32+zeile)+CHR$(32+spalte)
<32> 620 RETURN
<95> 630 *********************
<73> 640 *** Bild aufbauen ***
< 0> 650 *********************
<94> 660 PRINT cls$;
<67> 670 FOR i=1 TO 30
<42> 680 :. a=149
<59> 690 :. b=32
<76> 700 :. IF ((i MOD 6)=0) AND (i<30) THEN a=159
     :b=154
< 1> 710 :. c=b:d=b:e=b:f=b
<12> 720 :. IF ((i MOD 6)=3) AND (i<24) THEN c=49+
     INT(i/6):d=c+4:e=c+16+INT(i/6):f=e+1
< 6> 730 :. a$=STRING$(36,b)+CHR$(c)+CHR$(a)+CHR$(
     e)+STRING$(11,b)+CHR$(f)+CHR$(a)+CHR$(d)+STRI
     NG$(36,b)
<84> 740 :. PRINT a$
<50> 750 NEXT
< 3> 760 zeile=24
<11> 770 spalte=52
<11> 780 a$="Die Schwierige - eine Patience"
<50> 790 GOSUB 1920
<18> 800 a$="programmiert für"
< 6> 810 zeile=25
< 2> 820 spalte=52
<39> 830 GOSUB 1920
<16> 840 a$="CPC Amstrad International"
<26> 850 zeile=26
<48> 860 spalte=64
<47> 870 GOSUB 1920
<86> 880 a$="von"
<46> 890 zeile=27
<98> 900 spalte=52
<36> 910 GOSUB 1920
<46> 920 a$="Günter Kramer"
<47> 930 zeile=28
<59> 940 spalte=56
<44> 950 GOSUB 1920
<79> 960 a$="9"+SPACE$(4)+"0"
<67> 970 zeile=29
< 3> 980 spalte=42
<52> 990 GOSUB 1920
<78> 1000 RETURN
<41> 1010 *************************
<23> 1020 *** löschen Meldungen ***
<47> 1030 *************************
<27> 1040 a$=SPACE$(37)
<62> 1050 zeile=24:spalte=0
<96> 1060 GOSUB 1920
<80> 1070 PRINT
<67> 1080 FOR loesch=25 TO 28
<29> 1090 :. PRINT a$
<84> 1100 NEXT
< 3> 1110 PRINT a$;
<86> 1120 RETURN
<49> 1130 *******************
<29> 1140 *** Spiel läuft ***
<55> 1150 *******************
< 3> 1160 phase=1
< 0> 1170 FOR i=0 TO 12
< 9> 1180 :. FOR ablag=1 TO 8
<40> 1190 :. :. karte=folge(i*8+ablag-1)
<24> 1200 :. :. GOSUB 1970
<91> 1210 :. NEXT
<40> 1220 :. IF i=12 THEN 1300
<94> 1230 :. GOSUB 1040
<28> 1240 :. GOSUB 2560
< 1> 1250 :. GOSUB 1040
<24> 1260 :. IF b$=CHR$(13) THEN 1300
<37> 1270 :. GOSUB 2890
<30> 1280 :. IF b$=CHR$(13) THEN 1300
<67> 1290 :. GOTO 1230
<88> 1300 NEXT
<97> 1310 GOSUB 1040
<64> 1320 FOR i=1 TO 10
<22> 1330 :. zeile=26
<16> 1340 :. spalte=0
<77> 1350 :. a$="Alle Karten aufgelegt"
<17> 1360 :. GOSUB 1920
<67> 1370 :. PRINT CHR$(7)
<12> 1380 :. GOSUB 1040
<73> 1390 :. PRINT CHR$(7)
<90> 1400 NEXT
< 6> 1410 phase=2
< 9> 1420 GOSUB 2560
< 6> 1430 GOSUB 1040
<14> 1440 IF b$=CHR$(13) THEN 1600
< 0> 1450 GOSUB 2890
<20> 1460 IF b$=CHR$(13) THEN 1600
<35> 1470 IF geschafft<8 THEN GOTO 1420
<25> 1480 zeile=13
< 1> 1490 spalte=0
<84> 1500 a$=CHR$(134)+STRING$(25,138)+CHR$(140)
<91> 1510 GOSUB 1920
<75> 1520 PRINT
<55> 1530 PRINT CHR$(133)" Herzlichen Glückwunsch!
      "CHR$(133)
<51> 1540 PRINT CHR$(131)STRING$(25,138)CHR$(137)
<14> 1550 GOSUB 1040
<59> 1560 zeile=25
<96> 1570 spalte=0
<24> 1580 a$=esc$+"r"+"Weiter mit beliebiger Taste
      "+esc$+"u"
<16> 1590 GOSUB 1920
<90> 1600 RETURN
<62> 1610 **********************
<49> 1620 *** Karte ausgeben ***
<68> 1630 **********************
<34> 1640 farb$=CHR$(160+INT(karte/13))
```

```
<44> 1650 IF karte<26 THEN PRINT invers$
<72> 1660 a=1+(karte MOD 13)
<95> 1670 bild$=CHR$(a+48)
<66> 1680 IF a=10 THEN bild$="X"
< 4> 1690 IF a=11 THEN bild$="B"
<43> 1700 IF a=12 THEN bild$="D"
<42> 1710 IF a=13 THEN bild$="K"
<46> 1720 IF a=1 THEN bild$="A"
<73> 1730 a$=CHR$(150)+bild$+STRING$(1,154)+bild$+
     CHR$(156)
< 5> 1740 GOSUB 1920
<46> 1750 zeile=zeile+1
<19> 1760 a$=CHR$(149)+farb$+SPACE$(1)+farb$+CHR$(
     149)
<14> 1770 GOSUB 1920
<55> 1780 zeile=zeile+1
<35> 1790 a$=CHR$(149)+CHR$(32)+farb$+CHR$(32)+CHR
     $(149)
<94> 1800 GOSUB 1920
<36> 1810 zeile=zeile+1
<73> 1820 a$=CHR$(149)+SPACE$(3)+CHR$(149)
< 4> 1830 GOSUB 1920
<45> 1840 zeile=zeile+1
<19> 1850 a$=CHR$(147)+STRING$(3,154)+CHR$(153)
<13> 1860 GOSUB 1920
<83> 1870 PRINT normal$
<19> 1880 RETURN
<81> 1890 ******************
<65> 1900 *** Bildausgabe ***
<59> 1910 ******************
<95> 1920 PRINT home$;FNcursor$(zeile,spalte);a$;
< 6> 1930 RETURN
<32> 1940 *********************
<61> 1950 *** Karte ablegen ***
<38> 1960 *********************
<87> 1970 IF ablag<9 THEN GOSUB 2120
< 5> 1980 IF ablag>8 THEN GOSUB 2030
<24> 1990 RETURN
<12> 2000 ***************************************
<62> 2010 *** Karte auf Stapel oder in Keller ***
<18> 2020 ***************************************
<89> 2030 IF ablag>10 THEN ablag=ablag-10
<17> 2040 zeile=6*INT((ablag-1)/2)
<40> 2050 spalte=((ablag-1) MOD 2)*6+39
<86> 2060 GOSUB 1640
<96> 2070 ablage(ablag)=karte
< 4> 2080 RETURN
<75> 2090 **********************
<37> 2100 *** Karte in Reihe ***
<53> 2110 **********************
<30> 2120 anzahl=legen(ablag,0)+1
< 4> 2130 legen(ablag,anzahl)=karte
<94> 2140 zeile=6*((ablag-1) MOD 4)
< 0> 2150 spalte=33:zu=-2:pfeil=36
<89> 2160 IF ablag>4 THEN spalte=52:zu=2:pfeil=53
<68> 2170 legen(ablag,0)=anzahl
<19> 2180 legen(ablag,anzahl)=karte
<14> 2190 spalte1=spalte
<22> 2200 spalte=spalte+zu*anzahl
<61> 2210 IF anzahl<17 THEN GOSUB 1640:RETURN
<47> 2220 zeile1=zeile
<28> 2230 FOR ab=1 TO 16
<96> 2240 :. spalte=spalte1+ab*zu
<72> 2250 :. zeile=zeile1
<48> 2260 :. karte=legen(ablag,ab+anzahl-16)
< 5> 2270 :. GOSUB 1640
<12> 2280 NEXT
<62> 2290 spalte=pfeil
<58> 2300 zeile=zeile-1
<40> 2310 a$=CHR$(253)
< 6> 2320 IF ablag>4 THEN a$=CHR$(252)
< 4> 2330 IF anzahl=16 THEN a$=CHR$(32)
<97> 2340 GOSUB 1920
< 1> 2350 RETURN
<45> 2360 ******************
<57> 2370 *** Spielende. ***
<51> 2380 ******************
<68> 2390 b$=a$
<97> 2400 IF (phase=1) OR (block>0) THEN 2520
< 1> 2410 GOSUB 1040
<73> 2420 a$="Spielende?"
<49> 2430 zeile=25
<86> 2440 spalte=0
< 3> 2450 GOSUB 1920
<41> 2460 b$="1":a$=""
<61> 2470 WHILE b$="1"
<44> 2480 :. a$=UPPER$(INKEY$)
<28> 2490 :. IF a$="J" THEN b$=CHR$(13)
<92> 2500 :. IF a$="N" THEN b$=""
<25> 2510 WEND
<95> 2520 RETURN
<55> 2530 **********************
<74> 2540 *** Karte aufnehmen ***
<61> 2550 **********************
<18> 2560 GOSUB 1040
<45> 2570 a$="Bitte eine Karte aufnehmen"
<53> 2580 zeile=24
< 4> 2590 spalte=0
<91> 2600 GOSUB 1920
<75> 2610 PRINT
<69> 2620 PRINT "Eingeben: 1 - 8... = Reihe""
<10> 2630 PRINT SPC(10);"9 oder 0 = Keller""
<34> 2640 IF phase=1 THEN PRINT SPC(10);"RETURN..
     = weiter auflegen"
<62> 2650 IF phase=2 THEN PRINT SPC(10);"RETURN =
     Spielende"
<72> 2660 b$=""
<39> 2670 WHILE b$=""
<15> 2680 :. a$=INKEY$
<41> 2690 :. IF a$=CHR$(13) THEN 2390
<76> 2700 :. IF (a$>"/") AND (a$<":") THEN GOSUB 2
     760
<29> 2710 WEND
< 0> 2720 RETURN
<71> 2730 *********************************
< 7> 2740 *** Prüfen aufgenommene Karte ***
<77> 2750 *********************************
<52> 2760 aufnehmen=VAL(a$):von$=a$
<22> 2770 IF (aufnehmen=0) OR (aufnehmen=9) THEN 2
     810
<41> 2780 karte=legen(aufnehmen,legen(aufnehmen,0)
     )
<45> 2790 IF legen(aufnehmen,0)>0 THEN 2840 ELSE 2
     850
<53> 2800 :. IF a$=CHR$(13) THEN b$="OK"
<48> 2810 IF a$="0" THEN aufnehmen=10
<30> 2820 karte=ablage(aufnehmen)
<45> 2830 IF ablage(aufnehmen)>52 THEN 2850
<70> 2840 b$="OK"
<11> 2850 RETURN
<37> 2860 *********************
<66> 2870 *** Karte ablegen ***
<43> 2880 *********************
<78> 2890 a$="Wohin Karte aus Reihe "+von$+"?"
<61> 2900 block=0
<69> 2910 zeile=24:spalte=0
<64> 2920 GOSUB 1920:PRINT
<43> 2930 IF block=0 THEN PRINT "Eingeben: 0 - 9 =
     Reihe oder Keller"
<71> 2940 IF block>0 THEN PRINT "Eingeben: 1 - 8 =
     Reihe"
< 9> 2950 IF block=0 THEN PRINT SPC(10)"A - H = St
     apel"
<70> 2960 IF (block=0) AND (aufnehmen<9) AND (lege
     n(aufnehmen,0)>1) THEN PRINT SPC(10)"+.... =
     Block umlegen"
<97> 2970 IF phase=2 THEN PRINT SPC(10)"RETURN = n
     icht anlegen"
<95> 2980 IF phase=1 THEN PRINT SPC(10)"RETURN = w
     eiter auflegen"
<93> 2990 c$=""
<43> 3000 b$=""
<34> 3010 WHILE (b$="") OR (b$="1")
<58> 3020 :. IF b$<>"1" THEN a$=UPPER$(INKEY$) ELS
     E b$=""
<57> 3030 :. IF (a$>"/") AND (a$<":") AND (block=0
     ) THEN GOSUB 3170
<46> 3040 :. IF (a$>"0") AND (a$<"9") AND (block>0
     ) THEN GOSUB 4620
<39> 3050 :. IF (a$>"@") AND (a$<"I") AND (block=0
     ) THEN 3200
<59> 3060 :. IF (a$=CHR$(13)) AND (phase=1) THEN 2
     390
<80> 3070 :. IF aufnehmen>8 THEN 3110
<44> 3080 :. IF legen(aufnehmen,0)<2 THEN 3110
<76> 3090 :. IF ((a$=CHR$(22)) OR (a$=CHR$(43))) A
     ND (block=0) THEN GOSUB 4080
<64> 3100 IF c$="FE" THEN 2890
<75> 3110 :. IF b$="FE" THEN GOSUB 3870
<21> 3120 WEND
<91> 3130 RETURN
<18> 3140 ********************
<97> 3150 *** prüfen Ablage ***
<24> 3160 ********************
<34> 3170 ablag=VAL(a$)
<87> 3180 IF a$="0" THEN ablag=10
```

```
<82> 3190 GOTO 3210
< 7> 3200 ablag=ASC(a$)-54
<44> 3210 nach$=a$
<17> 3220 IF ablag>8 THEN 3290
<64> 3230 IF legen(ablag,0)=0 THEN 3390
<23> 3240 karte1=legen(ablag,legen(ablag,0))
<89> 3250 IF INT(karte/26)=INT(karte1/26) THEN 343
     0
<10> 3260 a=(karte1 MOD 13)-(karte MOD 13)
<46> 3270 IF (a=1) OR (a=-12) THEN 3390
<32> 3280 GOTO 3430
<73> 3290 IF ablag<11 THEN 3380
<96> 3300 IF ablage(ablag-10)>51 THEN 3360
<89> 3310 karte1=ablage(ablag-10)
<13> 3320 IF INT(karte/13)<>INT(karte1/13) THEN 34
     30
<85> 3330 a=(karte MOD 13)-(karte1 MOD 13)
< 7> 3340 IF (a=1) AND ((karte MOD 13)=12) THEN ge
     schafft=geschafft+1
< 1> 3350 IF a=1 THEN 3390 ELSE 3430
<38> 3360 IF (karte MOD 13)<>0 THEN 3430
<26> 3370 IF INT((ablag-11)/2)=INT(karte/13) THEN
     3390 ELSE 3430
<98> 3380 IF ablage(ablag)<51 THEN 3430
<76> 3390 b$="OK"
<59> 3400 GOSUB 1970
<57> 3410 IF block=0 THEN GOSUB 3480
<31> 3420 GOTO 3440
< 3> 3430 b$="FE"
< 1> 3440 RETURN
<27> 3450 *********************
<89> 3460 *** Karte löschen ***
<33> 3470 *********************
<11> 3480 IF aufnehmen>8 THEN 3770
<58> 3490 a=legen(aufnehmen,0)-1
< 4> 3500 zeile=6*INT((aufnehmen-1) MOD 4)
<51> 3510 zeile1=zeile
<44> 3520 spalte=33
<42> 3530 zu=-2
<38> 3540 pfeil=36
< 7> 3550 IF aufnehmen>4 THEN spalte=52:zu=2:pfeil
     =53
<15> 3560 legen(aufnehmen,0)=a
<17> 3570 spalte1=spalte
<67> 3580 spalte=spalte+zu*a
<22> 3590 ablag=aufnehmen
<27> 3600 anzahl=legen(ablag,0).................
     .......................
<87> 3610 IF a>15 THEN 2220
<64> 3620 spalte=spalte+zu
<68> 3630 FOR zeile=zeile1 TO zeile1+4
<29> 3640 :. a$=SPACE$(5)
<22> 3650 :. GOSUB 1920
<15> 3660 NEXT
<25> 3670 IF a=0 THEN 3830
<67> 3680 zeile=zeile1
<24> 3690 spalte=spalte-zu
<48> 3700 karte=legen(aufnehmen,a)
<86> 3710 GOSUB 1640
<55> 3720 a$=CHR$(32)
<76> 3730 zeile=zeile-1
<58> 3740 spalte=pfeil
<10> 3750 GOSUB 1920
<84> 3760 GOTO 3830
<59> 3770 ablage(aufnehmen)=99
<67> 3780 zeile1=24
<54> 3790 spalte=39
<89> 3800 IF aufnehmen=10 THEN spalte=45
<75> 3810 a=0
<50> 3820 GOTO 3630
< 6> 3830 RETURN
<32> 3840 *********************
<16> 3850 *** Fehlermeldung ***
<38> 3860 *********************
<70> 3870 zeile=25
<60> 3880 spalte=0
<60> 3890 a$=SPACE$(37)
<98> 3900 GOSUB 1920
<91> 3910 PRINT:PRINT a$:PRINT a$:PRINT a$
<58> 3920 a$="von "+von$+" nach "+nach$+" nicht mö
     glich"
<73> 3930 zeile=26
<97> 3940 spalte=0
<14> 3950 GOSUB 1920
<97> 3960 PRINT
<42> 3970 PRINT "neues Ziel eingeben, oder RETURN"
<25> 3980 b$="2"
<94> 3990 WHILE b$="2"
```

```
<14> 4000 :. a$=UPPER$(INKEY$)
<43> 4010 :. IF a$=CHR$(13) THEN b$="OK":a$=""
<72> 4020 :. IF a$<>"" THEN b$="1"
<23> 4030 WEND
<93> 4040 RETURN
<53> 4050 **********************
<60> 4060 *** Block auswählen ***
<59> 4070 **********************
<76> 4080 block=legen(aufnehmen,0)-1
<34> 4090 anzahl=block
<71> 4100 ende=anzahl-15
< 0> 4110 IF ende<2 THEN ende=2
< 0> 4120 GOSUB 1040
<83> 4130 a$="Reihe "+von$+" bis:"
<35> 4140 zeile=24
<85> 4150 spalte=0
< 2> 4160 GOSUB 1920
<85> 4170 PRINT
<35> 4180 PRINT "+ = vorwärts"
<47> 4190 PRINT "- = zurück"
<52> 4200 PRINT "RETURN = ablegen"
<57> 4210 c$=""
<77> 4220 a$="1"
<33> 4230 WHILE c$=""
< 3> 4240 :. karte=legen(aufnehmen,block)
<96> 4250 :. zeile=24
< 1> 4260 :. spalte=18
< 0> 4270 :. IF a$="" THEN 4360
<63> 4280 :. IF a$="-" THEN 4330
<86> 4290 :. karte1=legen(aufnehmen,block+1)
<92> 4300 :. IF INT(karte/26)=INT(karte1/26) THEN
     block=block+1:GOTO 4340
<64> 4310 :. a=(karte MOD 13)-(karte1 MOD 13)
<74> 4320 :. IF (a<>1) AND (a<>-12) THEN block=blo
     ck+1:GOTO 4340
<96> 4330 :. GOSUB 1640
<61> 4340 :. IF a$="1" THEN c$="FE":GOTO 4480
<88> 4350 :. a$=""
< 5> 4360 :. a$=INKEY$
<52> 4370 :. IF a$=CHR$(13) THEN c$="OK":GOTO 4480
<92> 4380 :. IF (a$=CHR$(22)) OR (a$=CHR$(43)) THE
     N 4410
<10> 4390 :. IF (a$=CHR$(28)) OR (a$=CHR$(45)) THE
     N 4450
< 1> 4400 :. GOTO 4480
<77> 4410 :. a$="+"
<86> 4420 :. IF block<ende THEN 4340
<24> 4430 :. block=block-1
<13> 4440 :. GOTO 4480
<88> 4450 :. IF block>anzahl THEN 4340
<19> 4460 :. a$="-"
<95> 4470 :. block=block+1
<46> 4480 WEND
<55> 4490 IF c$="FE" THEN GOSUB 1040:b$="OK":GOTO
     4580
<78> 4500 a$=SPACE$(16)
<47> 4510 zeile=25
<84> 4520 spalte=0
< 1> 4530 GOSUB 1920
<84> 4540 PRINT
<91> 4550 PRINT a$
<22> 4560 PRINT "in welche Reihe?"
<46> 4570 karte=legen(aufnehmen,block)
<16> 4580 RETURN
< 6> 4590 **********************
<19> 4600 *** umlegen im Block ***
<83> 4610 **********************
<89> 4620 GOSUB 3170
<26> 4630 IF b$="FE" THEN 4810
<20> 4640 ende=block+1
<79> 4650 IF ende>legen(aufnehmen,0) THEN 4710
<29> 4660 a$=nach$
<21> 4670 karte=legen(aufnehmen,ende)
< 8> 4680 GOSUB 3170
<62> 4690 ende=ende+1
<80> 4700 GOTO 4650
<59> 4710 ende=legen(aufnehmen,0)-block
<91> 4720 IF ende>15 THEN 4780
<17> 4730 FOR ende=legen(aufnehmen,0) TO block STE
     P-1
<47> 4740 :. legen(aufnehmen,0)=ende
<77> 4750 :. GOSUB 3480
<18> 4760 NEXT
<60> 4770 GOTO 4800
<35> 4780 legen(aufnehmen,0)=anzahl+1
<66> 4790 GOSUB 3480
<61> 4800 block=0
< 1> 4810 RETURN
```



Preiswert geplottet

Tschechischer A4-XY-Plotter für den CPC

Es ist früher Morgen, und man ist sich noch gar nicht so sicher, daß man wirklich arbeiten will. In diesem Stimmungstief ereilt uns die Nachricht, daß ein Paket angekommen ist. Als Absender finden wir eine Prager Adresse. Wir öffnen das Paket, und zum Vorschein kommt, wir trauen unseren Augen kaum, ein Plotter.

Plotter gehören bei den CPC-Nutzern zu den seltener genutzten Ausgabemedien. Auch die gängigen Programme unterstützen diese Geräte selten. Obwohl die Ausgabe von Texten auf einem Plotter möglich ist, kann er in dieser Beziehung kaum mit einem Drucker mithalten. Die Qualität von Strichgrafiken ist aber auf diesem Ausgabegerät unübertroffen oder kann nur mit einem ungleich höheren finanziellen Aufwand für einen Drukker erreicht werden. Das hier vorgestellte Gerät, den Plotter XY 4150, kann man nebst zugehöriger CPC-Software bereits für unter 200 DM bekommen. Hergestellt wird er von der Laboratorní prístroje Praha GmbH in Prag. Konzipiert ist dieser Plotter eigentlich für die Rechnertypen Atari 130/800, C 64/128 und ZX Spectrum, für die es ein spezielles Interface zur Ansteuerung des Gerätes gibt. Die Firma Unisoft in Prag nimmt an den Plottern eine geringfügige technische Änderung vor, wodurch der XY 4150 dann ohne ein spezielles Interface direkt am Centronics-Ausgang des CPC betrieben werden kann. Von dieser Firma stammt auch die mitgelieferte Software. Sie besteht aus RSX-Erweiterungen, die gleichnamig mit den BASIC-Grafikbefehlen des CPC sind.

Zusätzlich gibt es noch einige RSX-Erweiterungen zur Ausgabe und Beeinflussung (Winkel und Zeichenabstand) von Schrift und zur Erstellung von Grafikzeichen zur Ausgabe auf den Plotter. Im Handbuch sind alle zusätzlichen Befehle ausführlich erklärt. Hier erfährt man auch einiges über die technischen Daten und die Funktion des Gerätes. Zum Lieferumfang gehören vier Stifte mit den Farben Blau, Rot, Schwarz und Grün. Die jeweils gewünschte Zeichenfarbe wird per Hand eingelegt.

Steuerung per RSX

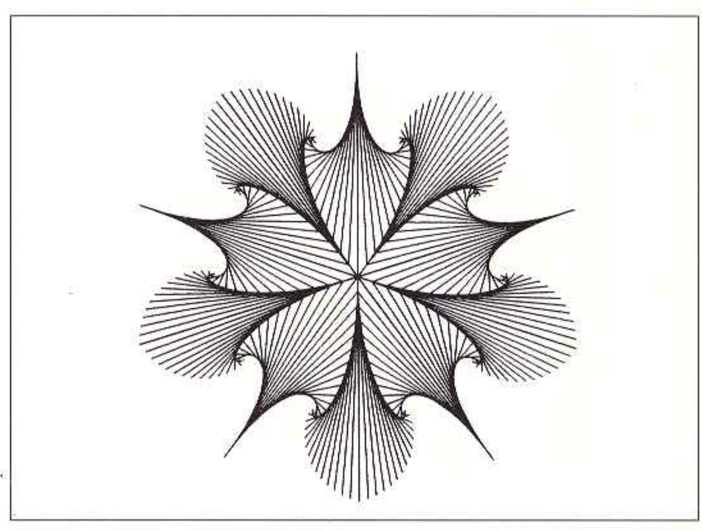
Der Plotter enthält keinerlei intelligente Elektronik. Alle Steueraufgaben muß der CPC übernehmen. Die Umwandlung der Grafikbefehle in die Steuerimpulse übernehmen die RSX-Routinen.

Die mitgelieferte Gebrauchsanweisung beschreibt zwar eingehend die RSX-Erweiterung und schneidet das Thema Hardware an, aber beim erstmaligen Aufbau und bei der Einjustierung fühlt man sich im Stich gelassen. So bedarf es einiger Fantasie, um den Stift ordnungsgemäß einzubauen. Auch die Justierung der Stifthöhe und des Stiftandrucks erfordert etwas Geduld.

Bei der Justierung geht man am besten folgendermaßen vor:

1. Stift (Anzeige PEN) ausschalten
2. Stift so weit heraus- oder hereindrehen, bis er deutlich vom Papier abhebt
3. Regler PEN FORCE an den linken Anschlag drehen (=Stiftandruck auf Minimum)
4. Demo-Programm starten
5. Regler PEN FORCE so weit nach rechts drehen, bis die Zeichnung in guter Qualität dargestellt wird

Haben Sie alles richtig aufgebaut und eingestellt, besitzen Sie mit dem Plotter XY 4150 ein vorzügliches Ausgabege-



Eine mit dem Plotter dargestellte Grafik

rät für grafische Darstellungen. Für den BASIC-Programmierer bieten sich hier Darstellungen mathematischer Funktionen, beispielsweise das Apfelmännchen, an. Eine mit dem Plotter erstellte Grafik sehen Sie in der Abbildung. Die Qualität der Grafikausgaben steht der auf guten Druckern in nichts nach. Haben Sie etwas mehr Zeit, können Sie sogar Listings auf dem Plotter ausgeben lassen. Hierfür steht eine RSX-Routine zur Verfügung, die alle Bildschirmausgaben auf den Plotter umleitet.

Beschäftigt man sich eine Weile mit dem Plotter, so vermißt man sehr schnell ein passendes Grafikprogramm. Denn ein Bild nur mittels BASIC-Befehlen darstellen zu lassen, ist sehr umständlich. Deshalb haben wir für Sie das Programm PLGRAF.BAS entwickelt. Es soll den Grundstock für ein Vektorgrafikprogramm darstellen. Um das Listing möglichst klein zu halten, ist es zwar von der optischen Aufmachung nicht sehr gut ausgefallen, aber es ist voll funktionsfähig.

Klein, aber fein

Das Programm ist vollständig in BASIC geschrieben. Lediglich zur Ausgabe auf den Plotter werden die RSX-Befehle des Plotterprogramms verwendet, das Sie beim Kauf des Druckers erhalten. Falls Sie keinen Plotter besitzen, können Sie auch statt der Plotterausgaberoutine eine Hardcopyroutine einbinden. Das Programm ist gut strukturiert, wodurch es Ihnen leichtgemacht wird, eigene Änderungen und Erweiterungen am Programm vorzunehmen. So kann man noch eine Routine zur Fehlerbehandlung oder zur Nutzung von Schrift im Bild einfügen.

Folgende Möglichkeiten bietet das Programm:

- Zeichnen von Linien
- Zeichnen von Rechtecken
- Zeichnen von Kreisen
- vergrößerte oder verkleinerte Ansicht von Bildausschnitten
- Vergrößern oder Verkleinern von ausgewählten Objekten
- Verschieben von ausgewählten Objekten
- Löschen von ausgewählten Objekten
- Retten oder Laden von Bildern im PLGRAF-Format

Die Bedienung des Programms erfolgt per Tastatur, Maus oder Joystick. In der obersten Bildschirmzeile wird das jeweils aktuelle Menü angezeigt. Der gewünschte Menüpunkt kann durch Bewegung eines Balkens oder durch die Betätigung der in Klammern dargestellten Hotkeys angewählt werden. Die Aktivierung erfolgt durch Betätigen der <ENTER>-Taste oder des Feuerknopfes. Um grafische Objekte darzustellen, werden jeweils zwei Punkte markiert:

Objekt	zu setzende Punkte
Linie	Anfang, Ende
Rechteck	Diagonale
Kreis	Mittelpunkt, Radius

Die gesetzten Punkte werden mit der Leertaste bestätigt.

Um Objekte zu markieren, wird mit den Cursortasten <Auf> und <Ab> das gewünschte Objekt angewählt (es wird andersfarbig dargestellt). Zur Bestätigung der Wahl wird ebenfalls die Leertaste gedrückt. Anschließend wird ein Menü angezeigt, in dem die gewünschte Funktion gewählt werden kann. Beim Verschieben oder Kopieren wird noch die Eingabe von neuen Koordinaten verlangt. Diese werden ebenfalls mit den Cursortasten ausgewählt und mit der Leertaste bestätigt.

Zum Laden oder Speichern einer Grafik wählt man den Menüpunkt Datei. Hier kann der gewünschte Dateiname entweder mittels Rollbalken ausgewählt oder direkt eingegeben werden. Als Typ ist die Erweiterung ".PLG" voreingestellt.

Das Programm legt die Daten der jeweiligen Grafik im Speicher ab der Adresse &8000 ab. Sind keine Programme an die obere Speichergrenze (wozu auch der Plottertreiber gehört) geladen, ist im Speicher Platz für 1094 Elemente. Mit geladenem Plottertreiber steht noch Speicher für zirka 455 Elemente zur Verfügung.

Und nun viel Spaß mit dem Plotter, falls Sie ihn sich kaufen, und unserem kleinen Programm. Vielleicht schicken Sie uns einmal Ihre Verbesserung und Weiterentwicklung des Grafikprogramms? jg

Technische Daten:

verwendetes Papierformat	DIN A4, 279x210 mm
Schreibfläche	260x185 mm 2600x1850 programmierbare Schritte
Größe eines Schrittes	0,1 mm
maximale Schreibgeschwindigkeit	1000 mm/s
Zeichen	ASCII und selbst definierbare Zeichen
Genauigkeit	+/- 0,8 Prozent
Schreibfeder	Schreibfeder KIN 0581, 0582 (blau, rot, schwarz, grün) Röhrchenfeder 0,35 bis 1 mm
Schreibmedium	Papier 60-90 g/m,,h2, Folie A4, 0,15 mm
Versorgungsspannung	220 V
Abmessungen	396x178x80 mm
Gewicht	2,8 kg
Preis	etwa 200 DM
zur Verfügung gestellt durch	Unisoft Vaveriova 54 CS-150 00 Praha 5

```
100 ' PLGRAF.BAS                              [492]
110 ' CPC 664/6128(Plus)                     [1532]
120 ' (c) 1992 CPC International (jg)        [1620]
130 '                                         [117]
140 MODE 1:PRINT:PRINT:PRINT"Plottertreibe [7159]
r einbinden?(j/n)":CLEAR INPUT
150 i$=INKEY$:IF i$="" THEN 150              [1374]
160 IF i$="j" THEN MEMORY &8FFF:LOAD"plott [5884]
er.bin",&9000:CALL &9000:fpl=1:ELSE fpl=0
170 hm=HIMEM-2:MEMORY &7FFF                  [2034]
180 GOSUB 350                                 [982]
190 mz=1:GOSUB 1010                           [635]
200 IF i=1 THEN GOSUB 1800:GOTO 190:'Datei [2620]
210 IF i=2 THEN GOSUB 2630:GOTO 190:'Zoom  [2383]
220 IF i=3 THEN GOSUB 250:GOTO 190:'edit   [1364]
```

```
230 IF i=4 THEN GOSUB 2280:GOTO 190:'Plott [2343]
240 GOTO 190                                  [407]
250 mz=2:GOSUB 1010                          [1031]
260 IF i=4 THEN GOSUB 2730:RETURN:'Markier [2671]
en
270 GOSUB 480                                 [885]
280 LOCATE#1,2,1:PRINT#1,x;" ";y;            [2179]
290 xa=x:ya=y:fo=f                           [1592]
300 GOSUB 480                                 [885]
310 fo=f                                      [283]
320 GOSUB 1240:GOSUB 750                     [1156]
330 CLS#1                                     [373]
340 GOTO 190                                  [407]
350 '                                         [117]
360 'Initialisierung                         [1300]
370 '                                         [117]
```

```
380 MODE 1                                         [506]
390 DEFINT x,y,v,h,o,f,i,b,l,m,n                   [2120]
400 IF fpl=1 THEN |INIT                            [1613]
410 bmax=hm:bpos=&8002:ff=4                        [575]
420 xmax=639:ymax=380:ymin=0:xmin=0:mz=1:m [3519]
lang=9
430 x=0:y=0:yo=y:xo=x:v=1:h=0:o=8:f=1:fo=h [2837]
440 PLOT x,y,v                                     [832]
450 WINDOW#1,1,40,1,1                              [1375]
460 zf=1:INK 2,6:INK 3,26,0                        [2077]
470 RETURN                                         [555]
480 '                                              [117]
490 'Koordinateneingabe                            [1586]
500 '                                              [117]
510 WHILE NOT (INKEY(47) AND INKEY(76)):WE [3820]
ND
520 WHILE INKEY(47) AND INKEY(76)                  [1357]
530 IF INKEY(0)=0 AND y<ymax THEN y=y+1            [1476]
540 IF INKEY(72)=0 AND y<ymax THEN y=y+1           [1311]
550 IF INKEY(0)=32 AND y<ymax THEN y=y+o           [1651]
560 IF INKEY(72)=32 AND y<ymax THEN y=y+o          [1470]
570 IF INKEY(2)=0 AND y>ymin THEN y=y-1            [1225]
580 IF INKEY(73)=0 AND y>ymin THEN y=y-1           [1267]
590 IF INKEY(2)=32 AND y>ymin THEN y=y-o           [1470]
600 IF INKEY(73)=32 AND y>ymin THEN y=y-o          [2008]
610 IF INKEY(8)=0 AND x>xmin THEN x=x-1            [3175]
620 IF INKEY(74)=0 AND x>xmin THEN x=x-1           [2004]
630 IF INKEY(8)=32 AND x>xmin THEN x=x-o           [1695]
640 IF INKEY(74)=32 AND x>xmin THEN x=x-o          [1022]
650 IF INKEY(1)=0 AND x<xmax THEN x=x+1            [1789]
660 IF INKEY(75)=0 AND x<xmax THEN x=x+1           [1004]
670 IF INKEY(1)=32 AND x<xmax THEN x=x+o           [1659]
680 IF INKEY(75)=32 AND x<xmax THEN x=x+o          [1393]
690 LOCATE#1,30,1:PRINT#1,x;" ";y;                 [1708]
700 PLOT xo*zf,yo*zf,fo:fo=TEST(x*zf,y*zf) [5566]
:IF fo=h THEN PLOT x*zf,y*zf,v:ELSE PLOT x
*zf,y*zf,h
710 xo=x:yo=y                                      [396]
720 WEND                                           [390]
730 CLEAR INPUT                                    [184]
740 RETURN                                         [555]
750 '                                              [117]
760 '1 Teil Zeichnen: PE-xa,ya,x,y,i$              [2137]
770 '                                              [117]
780 ON i GOTO 810,840,860                          [1158]
790 PLOT xa*zf,ya*zf,h:PLOT x*zf,y*zf,h:fo [2915]
=h
800 RETURN                                         [555]
810 PLOT xa*zf,ya*zf,f:DRAW x*zf,y*zf,f            [1982]
820 RETURN                                         [555]
830 '                                              [117]
840 PLOT xa*zf,ya*zf,f:DRAW x*zf,ya*zf,f:D [5447]
RAW x*zf,y*zf,f:DRAW xa*zf,y*zf,f:DRAW xa*
zf,ya*zf,f
850 RETURN                                         [555]
860 '                                              [117]
870 'Kreis                                         [373]
880 xaz=xa*zf:xd=(x-xa)*zf                         [1586]
890 yaz=ya*zf:yd=(y-ya)*zf                         [2367]
900 xr=SQR(xd*xd+yd*yd)                            [1570]
910 rq=xr*xr:xx=0:xd=1                             [640]
920 WHILE xx<xd                                    [1170]
930 xd=SQR(rq-xx*xx):xx=xx+2                       [2788]
940 PLOT xaz+xd,yaz+xx,f:PLOT xaz-xd,yaz+x [2913]
x,f
950 PLOT xaz+xd,yaz-xx,f:PLOT xaz-xd,yaz-x [2927]
x,f
960 PLOT xaz+xx,yaz+xd,f:PLOT xaz-xx,yaz+x [3962]
d,f
970 PLOT xaz+xx,yaz-xd,f:PLOT xaz-xx,yaz-x [3406]
d,f
980 WEND                                           [390]
990 PLOT xa,ya,h                                   [827]
1000 RETURN                                        [555]
1010 '                                             [117]
1020 'Menueabfrage                                 [2037]
1030 '                                             [117]
1040 IF mz=1 THEN RESTORE 2250                     [1249]
1050 IF mz=3 THEN RESTORE 2270                     [1002]
1060 a$="":READ mxc                                [1381]
1070 FOR xc=1 TO mxc:READ n$(xc):NEXT              [1469]
1080 FOR xc=1 TO mxc:READ nn$(xc):NEXT             [1293]
1090 FOR xc=1 TO mxc:READ n(xc):NEXT               [1852]
1100 CLS#1                                         [373]
1110 LOCATE#1,1,1:FOR xc=1 TO mxc:PRINT#1, [4414]
n$(xc);:NEXT
1120 flag=0:xc=1:yc=1                              [888]
1130 GOSUB 2140                                    [863]
1140 CLEAR INPUT                                   [184]
1150 WHILE flag=0                                  [837]
1160 i$=INKEY$                                     [1004]
1170 FOR xcc=1 TO mxc                              [657]
1180 IF i$=nn$(xcc) THEN i=n(xcc):flag=1           [2647]
1190 NEXT                                          [350]
1200 GOSUB 2030                                    [877]
1210 WEND                                          [390]
1220 CLS#1                                         [373]
1230 RETURN                                        [555]
1240 '                                             [117]
1250 'Bildspeicherung                              [2623]
1260 '                                             [117]
1270 POKE bpos,i:bpos=bpos+1:GOSUB 1350            [2891]
1280 POKE bpos,(xa MOD 255):POKE bpos+1,(I [4056]
NT(xa/255)):bpos=bpos+2:GOSUB 1350
1290 POKE bpos,(ya MOD 255):POKE bpos+1,(I [3214]
NT(ya/255)):bpos=bpos+2:GOSUB 1350
1300 POKE bpos,(x MOD 255):POKE bpos+1,(IN [3203]
T(x/255)):bpos=bpos+2:GOSUB 1350
1310 POKE bpos,(y MOD 255):POKE bpos+1,(IN [3719]
T(y/255)):bpos=bpos+2:GOSUB 1350
1320 POKE &8000,((bpos-&8000) MOD 255)             [1377]
1330 POKE &8001,(INT((bpos-&8000)/255))            [2250]
1340 RETURN                                        [555]
1350 '                                             [117]
1360 'Speicherkontrolle                            [409]
1370 '                                             [117]
1380 IF bpos>bmax THEN flag=1:LOCATE 25,0: [3676]
PRINT CHR$(7);"Speicher voll!":ELSE flag=0
1390 RETURN                                        [555]
1400 '                                             [117]
1410 'Neuer Bildaufbau                             [570]
1420 '                                             [117]
1430 CLS                                           [91]
1440 ORIGIN xori,yori                              [900]
1450 bcount=&8002                                  [236]
1460 WHILE bcount<bpos                             [1526]
1470 i=PEEK(bcount):bcount=bcount+1                [2830]
1480 xa=PEEK(bcount)+PEEK(bcount+1)*255:bc [3460]
ount=bcount+2
1490 ya=PEEK(bcount)+PEEK(bcount+1)*255:bc [4746]
ount=bcount+2
1500 x=PEEK(bcount)+PEEK(bcount+1)*255:bco [2958]
unt=bcount+2
1510 y=PEEK(bcount)+PEEK(bcount+1)*255:bco [4652]
unt=bcount+2
1520 GOSUB 750                                     [935]
1530 WEND                                          [390]
1540 x=xori/zf* -1:y=yori/zf* -1                   [1726]
1550 RETURN                                        [555]
1560 '                                             [117]
1570 'Abfrage des Dateinamens                      [914]
1580 '                                             [117]
1590 PRINT"D(i)rekteingabe";                       [1934]
1600 y=VPOS(#0)                                    [710]
1610 LOCATE 1,y+2                                  [1060]
1620 a$="*.plg":|DIR,@a$:a$=""                     [3023]
1630 yy=VPOS(#0)                                   [1459]
1640 x=1:y=y+5                                     [1254]
1650 LOCATE x,y                                    [672]
1660 FOR c=1 TO 12:a$=a$+COPYCHR$(#0):LOCA [3536]
TE x+c,y:NEXT
1670 CALL &BB9C:LOCATE x,y:PRINT a$;:CALL  [5597]
&BB9C:LOCATE x,y
1680 IF INKEY(0)=0 AND y>1 THEN y=y-1:GOTO [1783]
 1750
1690 IF INKEY(2)=0 AND y<24 THEN y=y+1:GOT [536]
O 1750
1700 IF INKEY(1)=0 AND x<60 THEN x=x+15:GO [2498]
TO 1750
1710 IF INKEY(8)=0 AND x>15 THEN x=x-15:GO [3158]
TO 1750
1720 IF INKEY(18)=0 THEN 1760                      [335]
1730 IF INKEY(35)=0 THEN 1770                      [889]
1740 GOTO 1680                                     [395]
1750 PRINT a$;:a$="":GOTO 1650                     [756]
1760 RETURN                                        [555]
1770 CLEAR INPUT:LOCATE 1,yy:INPUT"Dateina [2266]
me:",a$
1780 IF INSTR(1,RIGHT$(a$,4),".")=0 THEN a [2878]
$=a$+".plg"
1790 RETURN                                        [555]
1800 '                                             [117]
1810 'Datei Ein- Ausgabe                           [1108]
1820 '                                             [117]
1830 CLS                                           [91]
1840 PRINT"(L)aden   (S)peichern":PRINT            [2925]
1850 CLEAR INPUT                                   [184]
1860 i$=INKEY$:IF i$="" THEN 1860                  [1492]
1870 IF i$="l" THEN 1900                           [951]
1880 IF i$="s" THEN 1970                           [574]
1890 CLS:RETURN                                    [832]
1900 GOSUB 1560                                    [899]
1910 CLS                                           [91]
1920 IF a$="" THEN RETURN                          [1147]
```

```
1930 LOAD a$,&8000                                    [650]
1940 bpos=PEEK(&8000)+PEEK(&8001)*255                 [1530]
1950 bpos=bpos+&8000                                  [671]
1960 GOTO 1400                                        [387]
1970 '                                                [117]
1980 GOSUB 1560                                       [899]
1990 CLS                                              [91]
2000 IF a$="" THEN RETURN                             [1147]
2010 SAVE a$,b,&8000,bpos-&8000                       [1723]
2020 GOTO 1400                                        [387]
2030 '                                                [117]
2040 'Menuepunktewahl                                 [2445]
2050 '                                                [117]
2060 IF INKEY(8)=0 AND xc>mlang THEN xc=xc            [4264]
-mlang:GOTO 2130
2070 IF INKEY(74)=0 AND xc>mlang THEN xc=x            [2644]
c-mlang:GOTO 2130
2080 IF INKEY(1)=0 AND xc<(mxc*mlang-mlang            [2584]
) THEN xc=xc+mlang:GOTO 2130
2090 IF INKEY(75)=0 AND xc<(mxc*mlang-mlan            [3383]
g) THEN xc=xc+mlang:GOTO 2130
2100 IF INKEY(18)=0 THEN GOTO 2180                    [728]
2110 IF INKEY(76)=0 THEN GOTO 2180                    [904]
2120 RETURN                                           [555]
2130 PRINT#1,a$;                                      [334]
2140 LOCATE#1,xc,yc                                   [415]
2150 a$="":FOR oc=1 TO mlang:a$=a$+COPYCHR            [3343]
$(#1):LOCATE#1,xc+oc,yc:NEXT
2160 LOCATE#1,xc,yc:PEN#1,0:PAPER#1,1:PRIN            [5102]
T#1,a$;:PEN#1,1:PAPER#1,0:LOCATE#1,xc,yc
2170 RETURN                                           [555]
2180 FOR xcc=1 TO mxc                                 [657]
2190 IF a$=n$(xcc) THEN i=n(xcc):flag=1               [2256]
2200 NEXT                                             [350]
2210 RETURN                                           [555]
2220 '                                                [117]
2230 'Menue-Datas                                     [1130]
2240 '                                                [117]
2250 DATA 4,"(D)atei   ","(Z)oom    ","(E)di          [4703]
t    ","(P)lott   ",d,z,e,p,1,2,3,4
2260 DATA 4,"(L)inie   ","(R)chtck  ","(K)re          [3131]
is   ","(M)ark    ",l,r,k,m,1,2,3,4
2270 DATA 4,"(S)kal    ","(L)oesch  ","(V)er          [5475]
sch  ","(K)opie   ",s,l,v,k,1,2,3,4
2280 '                                                [117]
2290 'Plotten                                         [1048]
2300 '                                                [117]
2310 CLS#1:PRINT#1,"Ausgabe auf den Plotte            [4764]
r?(j/n)";:CLEAR INPUT
2320 i$=INKEY$:IF i$="" THEN 2320                     [1496]
2330 IF i$<>"j" THEN RETURN                           [889]
2340 GOSUB 1410                                       [905]
2350 |INIT                                            [527]
2360 bcount=&8002                                     [236]
2370 WHILE bcount<bpos                                [1526]
2380 i=PEEK(bcount):bcount=bcount+1                   [2830]
2390 xa=(PEEK(bcount)+PEEK(bcount+1)*255)*            [5158]
2.89:bcount=bcount+2
2400 ya=(ymax-(PEEK(bcount)+PEEK(bcount+1)            [3916]
*255))*2.89:bcount=bcount+2
2410 x=(PEEK(bcount)+PEEK(bcount+1)*255)*2            [4163]
.89:bcount=bcount+2
2420 y=(ymax-(PEEK(bcount)+PEEK(bcount+1)*            [6487]
255))*2.89:bcount=bcount+2
2430 GOSUB 2470                                       [821]
2440 WEND                                             [390]
2450 |MOVE,0,0                                        [1101]
2460 RETURN                                           [555]
2470 '                                                [117]
2480 'Ausgabe auf Plotter                             [2030]
2490 '                                                [117]
2500 ON i GOTO 2520,2550,2570                         [901]
2510 RETURN                                           [555]
2520 |PLOT,xa,ya:|DRAW,x,y                            [1620]
2530 RETURN                                           [555]
2540 '                                                [117]
2550 |PLOT,xa,ya:|DRAW,x,ya:|DRAW,x,y:|DRA            [4484]
W,xa,y:|DRAW,xa,ya
2560 RETURN                                           [555]
2570 '                                                [117]
2580 'Kreis                                           [373]
2590 xd=x-xa:yd=y-ya:xr=SQR(xd*xd+yd*yd)              [3110]
2600 |MOVE,xa,ya                                      [1709]
2610 |CIRCLE,xr                                       [1273]
2620 RETURN                                           [555]
2630 '                                                [117]
2640 'Zoom                                            [837]
2650 '                                                [117]
2660 CLS#1:CLEAR INPUT:INPUT#1,"Faktor(Bil            [2892]
dschirm):";zfhilf
2670 IF zfhilf<=1 THEN xori=0:yori=0:GOTO             [3081]
2700
2680 CLS#1:PRINT"Neuer Nullpunkt:";:GOSUB             [2814]
480
2690 xori=x* -1*zfhilf:yori=y* -1*zfhilf              [2523]
2700 zf=zfhilf:GOSUB 1410                             [1365]
2710 CLS#1                                            [373]
2720 RETURN                                           [555]
2730 '                                                [117]
2740 '1 Teil markieren                                [1115]
2750 '                                                [117]
2760 bcount=&8002                                     [236]
2770 lflag=0                                          [220]
2780 WHILE bcount<bpos AND lflag=0                    [2485]
2790 obcount=bcount                                   [547]
2800 i=PEEK(bcount):ii=i:bcount=bcount+1              [3154]
2810 xa=PEEK(bcount)+PEEK(bcount+1)*255:bc            [3460]
ount=bcount+2
2820 ya=PEEK(bcount)+PEEK(bcount+1)*255:bc            [4746]
ount=bcount+2
2830 x=PEEK(bcount)+PEEK(bcount+1)*255:bco            [2958]
unt=bcount+2
2840 y=PEEK(bcount)+PEEK(bcount+1)*255:bco            [4652]
unt=bcount+2
2850 f=2:GOSUB 750                                    [1734]
2860 PLOT xa,ya,3                                     [568]
2870 CLEAR INPUT:WHILE INKEY(47) AND INKEY            [4948]
(2) AND INKEY(0):WEND
2880 f=1                                              [395]
2890 IF INKEY(0)=0 THEN IF bcount<&8014 TH            [5224]
EN bcount=obcount:ELSE bcount=bcount-18
2900 IF INKEY(47)=0 THEN lflag=1: ELSE lfl            [1171]
ag=0:GOSUB 750
2910 WEND                                             [390]
2920 IF lflag=0 THEN RETURN                           [616]
2930 mz=3:GOSUB 1010                                  [837]
2940 IF i=1 THEN GOSUB 2990:GOSUB 1410:RET            [2760]
URN
2950 IF i=2 THEN GOSUB 3120:GOSUB 1410:RET            [3356]
URN
2960 IF i=3 THEN GOSUB 3230:GOSUB 1410:RET            [1975]
URN
2970 IF i=4 THEN GOSUB 3300:GOSUB 1410:RET            [3219]
URN
2980 RETURN                                           [555]
2990 '                                                [117]
3000 'Skalieren                                       [418]
3010 '                                                [117]
3020 CLS#1:CLEAR INPUT:INPUT#1,"Skalierfak            [2801]
tor:";sf
3030 hilf=x:x=(xa+x+(x-xa)*sf)/2:xa=(xa+hi            [2676]
lf-(hilf-xa)*sf)/2
3040 hilf=y:y=(ya+y+(y-ya)*sf)/2:ya=(ya+hi            [2483]
lf-(hilf-ya)*sf)/2
3050 bcount=obcount                                   [585]
3060 POKE bcount,ii:bcount=bcount+1                   [1980]
3070 POKE bcount,(xa MOD 255):POKE bcount+            [4233]
1,(INT(xa/255)):bcount=bcount+2
3080 POKE bcount,(ya MOD 255):POKE bcount+            [4483]
1,(INT(ya/255)):bcount=bcount+2
3090 POKE bcount,(x MOD 255):POKE bcount+1            [4172]
,(INT(x/255)):bcount=bcount+2
3100 POKE bcount,(y MOD 255):POKE bcount+1            [4494]
,(INT(y/255)):bcount=bcount+2
3110 RETURN                                           [555]
3120 '                                                [117]
3130 'markiertes Teil loeschen                        [1775]
3140 '                                                [117]
3150 bpos=bpos-9                                      [651]
3160 WHILE obcount<bpos                               [1387]
3170 hilf=PEEK(bcount):POKE obcount,hilf              [3618]
3180 bcount=bcount+1:obcount=obcount+1                [2233]
3190 WEND                                             [390]
3200 POKE &8000,((bpos-&8000) MOD 255)                [1377]
3210 POKE &8001,(INT((bpos-&8000)/255))               [2250]
3220 RETURN                                           [555]
3230 '                                                [117]
3240 'markiertes Objekt verschieben                   [1853]
3250 '                                                [117]
3260 CLS#1:PRINT#1,"Neue Position:";                  [1248]
3270 xhilf=x-xa:yhilf=y-ya:GOSUB 480                  [2035]
3280 xa=x:x=xa+xhilf:ya=y:y=ya+yhilf                  [1335]
3290 GOTO 3050                                        [447]
3300 '                                                [117]
3310 '1 Objekt kopieren                               [2671]
3320 '                                                [117]
3330 CLS#1:PRINT#1,"Position:";                       [1989]
3340 xhilf=x-xa:yhilf=y-ya:x=x+4:y=y+4:GOS            [2743]
UB 480
3350 xa=x:x=xa+xhilf:ya=y:y=ya+yhilf                  [1335]
3360 i=ii:GOSUB 1240                                  [693]
3370 RETURN                                           [555]
```



Ein Plus für alle Fälle

CPC 2128: Plus-Power auch für die alten CPCs

4096 Farben, Cartrigeschacht, DMA-Sound, Hardwaresprites und noch so manches mehr haben die neuen CPCs an Bord. Dies gibt sicher Grund für Eifersüchteleien von seiten der Besitzer eines CPC der normalen Bauart. Trotzdem kein Grund zur Panik. Aus dem guten alten läßt sich in Windeseile ein neuer bauen.

Genau zur Jahreswende 1990/91 war es klar, die Firma Amstrad präsentierte ihre neuen CPCs. Wie auch bei den neuen PCWs hält sich die Amstrad Deutschland GmbH in dieser Richtung zurück und ist aus bestimmten Gründen nicht bereit, die neuen CPCs auf den deutschen Markt zu bringen.

Ein Gerücht kursiert

Ganz anders sieht es da schon bei unseren Nachbarn in England und Frankreich aus. Wilde Gerüchte kursieren, und so wird auch die CPC International immer heißer auf die neuen Geräte.

Kein Plus in Sicht

Doch, man soll es nicht glauben, nirgends ist ein solches Gerät zu bekommen. Umfangreiche Verhandlungen mit den englischen und französischen Amstrad-Niederlassungen machen es dann doch noch möglich, daß wir jeweils einen 464 als auch einen 6128 plus zu Gesicht bekommen. Und siehe da, nach einer abenteuerlichen Reise „Eschwege – London/Brentwood – Paris/Sevres – Eschwege" zierte dann doch endlich ein CPC 6128 Plus das Titelbild der CPC 12/1 '90/91.

Inzwischen sind schon fast zwei Jahre vergangen. Fast vergessen der Streß, und vor allem die Neugierde, was die neuen CPCs nun wirklich auf dem Kasten haben. Wahren Tüftlern verdanken wir nun, daß keines der Plus-Geheimnisse mehr geheim bleiben muß, lediglich eine Sache fehlte noch, ein Upgrade-Kit für Besitzer der alten CPC-Modelle. Doch auch dieses soll nicht lange auf sich warten lassen.



Das Herzstück des 2128: Die GX4000

GX4000 im Huckepack

Wer schon einen CPC der Baureihen 464, 664 oder 6128 besitzt, wird sicherlich vor der Anschaffung eines weiteren CPCs zurückschrecken. Will man doch das damals so teuer erworbene Gerät nicht unbedingt weit unter Preis verkaufen, um dann für den neuen noch eine ganze Stange Geld zu investieren. Außerdem gibt es noch ein weiteres Problem. Sind doch die Plus-Modelle nicht hundertprozentig zu den alten CPCs kompatibel. Das könnte also im schlimmsten Fall bedeuten, daß die vorhandene und eventuell auch häufiger eingesetzte Software nicht mehr auf den Neuen arbeitet. Ein großes Risiko also.

Auf verschlungenen Pfaden

Die NoName Service Gbr, Duisburg hat jetzt einen Umbau vollzogen, der selbst die Redaktion zum Staunen brachte. So wird angeboten, nach Einsendung der alten Computertastatur und des Floppycontrollers beim 464 das Gerät so aufzurüsten, daß es sowohl im normalen als auch im Plus-Modus funktioniert. Per Schalter kann dann nach dem rund 500 DM teuren Umbau zwischen beiden Modi umgeschaltet werden.

Erstmalig bestaunen konnten diesen Umbau die Besucher der CPC-Party in Reims, Frankreich. Mehr hierüber jedoch auf Seite 7.

Das Herz des uns vorliegenden Beta-Modells besteht aus einer GX4000-Konsole, die im Grunde genommen nichts anderes als ein abgespeckter CPC 464-Plus mit fehlenden Kabelverbindern für Tastatur, Floppy oder andere Hardware darstellt. Lediglich die P-Taste (Für Pause) wurde ihr noch gelassen. Weiterhin wurde die GX4000, die standardmäßig nur über 64 kByte Speicher verfügt, mit einer weiteren 64-kByte-Bank ausgerüstet.

6128-Plus kompatibel

Sämtliche Ausgänge wie Drucker- und Expansionsport sind direkt an der



Durch futuristisches Kurvendesign schlugen die neuen CPCs voll aus der Art. Trotzdem wurde auf Altbewährtes Wert gelegt: Floppy oder Tape sind im Tastaturgehäuse untergebracht.



Entgegen der Hardware in den alten CPCs wirkt die Floppy im Innenraum eines CPC 6128 (plus) im Vergleich zur Platine regelrecht monströs

GX4000 angebracht. Weiterer Vorteil des GX4000-Umbaus ist der entgegen den CPC-Plus-Modellen vorhandene Stecker zum Anschluß an ein handelsübliches Fernsehgerät mit Antenneneingang. Das Videointerface aus der CPC 8/9'92 muß also nicht erst nachgebaut werden.
Sozusagen als kleiner Bonus wurde der GX4000-Konsole noch ein Pauseschalter sowie ein Resettaster eingebaut.

Ein Stück mehr Kompatibilität

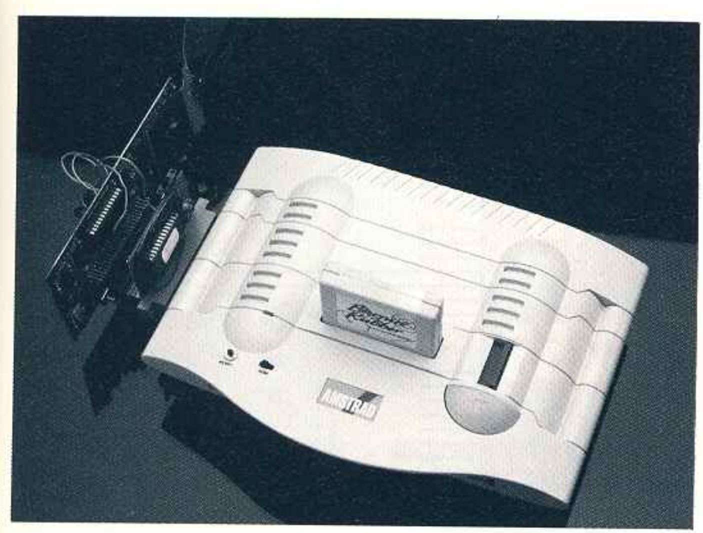
Lediglich beim Sound müssen Abstriche gemacht werden. Verfügen die Plus-Monitore über eingebaute Lautsprecher, wird bei dem CPC 2128 nur ein 3.5" Klinkenstecker angeboten, über den der Sound zum Beispiel an die Stereoanlage oder direkt an Aktivboxen gegeben werden kann.

Tips vom Tüftler

Alles in allem eine Menge Umbauarbeit, die Ihre 500,– DM sicherlich wert ist. Bei ausgiebigen Tests der Hardware mit Standardsoftware konnten keinerlei Fehler festgestellt werden. Verblüffend: Dadurch, daß der alte Diskettencontroller an dem Umbau weiterverwendet wird, läuft Software wie Discology oder Mastercopy, die auf den normalen Plus-Modellen nicht funktioniert, einwandfrei und das im Plus-Modus.
Auch dieser Umbau zeigt uns wieder, daß es gerade in Deutschland noch engagierte CPC-Tüftler gibt. Wir hoffen, daß dieser Umbau auch andere Leser motivierte ein wenig zu basteln.
Aber kurz noch ein kleines Schmankerl vorweg: In der nächsten CPC verrät der Bastler einige Tricks seiner Umbauaktion. Praktisch nebenbei hat er so herausgefunden, wie sich ein CPC 464 Plus mit 64 kByte in einen 464 Plus mit satten 128 kByte wandeln läßt und das wollen wir Ihnen natürlich nicht vorenthalten.

rs

Die Vorab-Version dieses CPC-Umbaus wurde uns freundlicherweise von der Firma
NoName EDV Service Gbr,
In den Peschen 51,
4100 Duisburg 14,
zur Verfügung gestellt.



Der Floppycontroller und das Zusatzrom für den normalen CPC-Betrieb sind direkt an der GX4000 angebracht



Mike Behrendt, Gründer der Noname Service Gbr sowie der Bastler Wolfgang Noisternig bescherten uns den CPC-Umbau

Wetterstation PCW

Temperaturdaten mit dem PCW auswerten

**Wer meint, daß Windmessung alles ist, was der Joyce kann, der sollte sich eines Besseren belehren lassen. Unter Verwendung von nur zwei Ports kann der Joyce nun auch Temperaturdaten erfassen und auswerten.
Bei minimalem Hard- und Softwareaufwand eine gelungene Lösung, die sich von jedem in die Praxis umsetzen läßt.**



Die Dateneingabe kann über dieses Meßmodul erfolgen

Nachdem das Windmodul in der CPC 8/9'92 veröffentlicht wurde, folgt nun noch ein Thermometer. Auch bei diesem Aufbau wurde wieder auf eine möglichst einfache und vor allem preisgünstige Lösung Wert gelegt. Es werden weder aufwendige Programme noch aufwendige Hardware benötigt. Ein recht preisgünstiges Modul (zum Beispiel über die Firma Conrad zu beziehen/siehe Bild) macht's möglich: Temperaturdaten mit dem PCW erfassen, speichern und auswerten.

Hardware für unter 40,– DM

Wie auch bei der Windmessungsanlage wird davon ausgegangen, daß eine programmierbare Schnittstelle nach Bernhard Graßhoff (Autor JOYCE – Mehr als ein Textsystem) und das oben angegebene Temperaturmeßmodul vorhanden sind. Dieses Modul besitzt einen seriellen Ausgang, der bitweise ein- und ausgelesen wird. Da dieser Datenausgang jedoch keinen TTL-Pegel liefert, ist noch ein kleiner Pegelwandler nach Skizze 2 nötig. Er läßt sich jedoch leicht auf einer Lochrasterplatine mit Bauteilen aus der Bastelkiste aufbauen. Skizze 2 liefert die benötigten Informationen dazu.
Alles in allem wird der Geldbeutel mit weniger als 40,– DM belastet, vorausgesetzt, die Schnittstelle ist bereits vorhanden. Wer sich die Schnittstelle sparen will, kann natürlich auch Daten über den Druckerport einlesen. Hier bietet sich die Adresse FDh (Busy und Paper Out on). Dann ist allerdings der Drucker nicht gleichzeitig nutzbar.

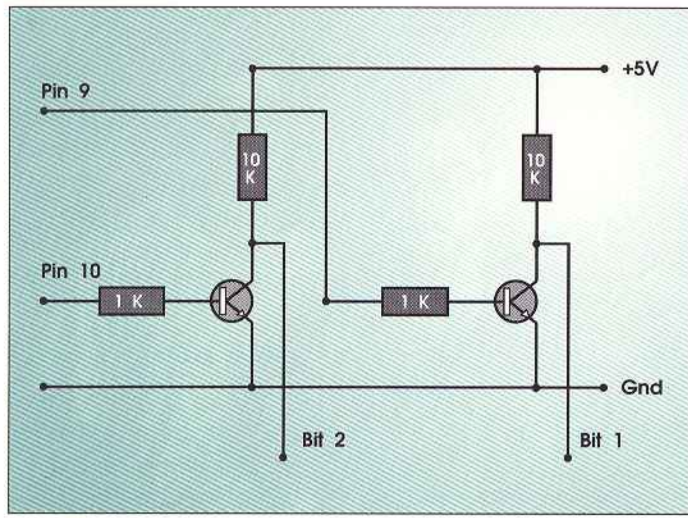
Software: Ein paar Programmzeilen genügen

Die zu verarbeitenden Daten stehen an Pin 9 und der Takt an Pin 10 des Meßmoduls im Abstand von einer beziehungsweise zehn Sekunden bereit (Skizze 1). Die Datenübertragung selbst dauert etwa 18 ms, so daß Joyce sich ganz schön sputen muß, um alle Daten zu erwischen. Aus diesem Grund werden die Daten erst ausgelesen, in einem Array zwischengespeichert und dann in der Pause ausgewertet.

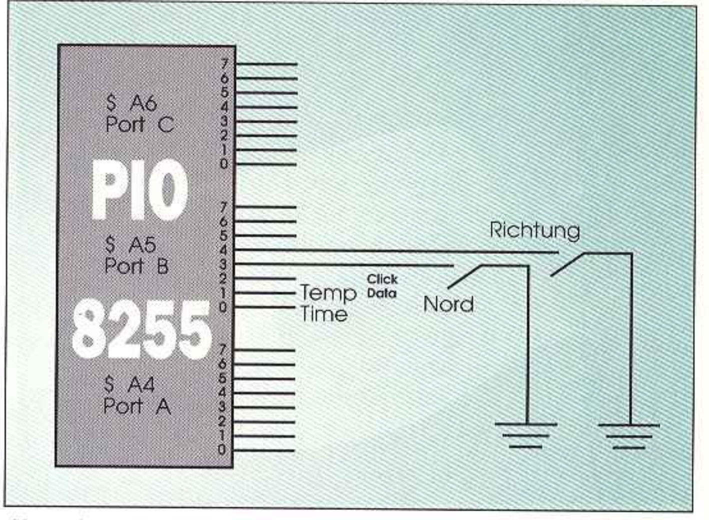
Die Prozedur Messen erledigt diese Aufgabe. Sie fragt den Port A5h ständig ab und wartet auf den High Pegel an Pin 10. Dann werden die folgenden Daten in das Array Data eingelesen. 200 Samples reichen bei dem gemächlichen Tempo des Joyce vollkommen für eine Meßperiode aus. Während des Meßvorgangs müssen die Interrupts ausgeschaltet sein, sonst kommt es zu Fehlmessungen. Die Prozedur Auswerten wandelt die Daten in einen Temperaturwert um. Der Temperaturwert selbst ist als 13-Bit-Wort nach folgendem Schema verschlüsselt:

D1	Vorzeichen
D2 -D5	Zehner
D6 -D9	Einer
D10-D13	Zehntel

Der errechnete Wert wird der Integervariablen temp in 1/10 Grad zugeordnet. Der Einfachheit halber wurde diese Variable global deklariert.
Die Temperaturwerte können nun beliebig weiterverarbeitet werden.



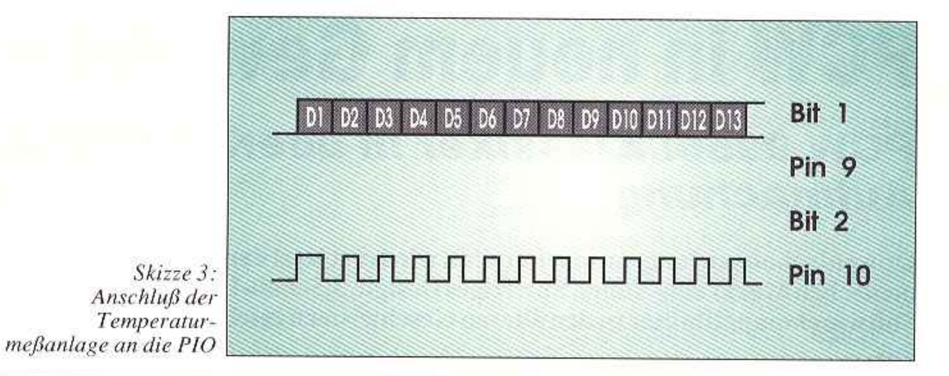
Skizze 1: An Pin 9 stehen die Daten und an Pin 10 der Takt bereit. Dies erfolgt jede beziehungsweise alle zehn Sekunden



Skizze 2: Diese kleine Schaltung ist leicht auf einer Lochrasterplatine nachzuvollziehen

Für die Auswertung in einer Wetterstation kann die Aufzeichnung über eine Zeitachse recht anschaulich sein.
Mit den Prozeduren der Grafik-Toolbox aus dem Joyce Sonderheft 2, ist eine einfache Grafikauswertung problemlos möglich.
Um das Programmlisting jedoch möglichst kurz und vor allem übersichtlich zu halten, wurde allerdings hier auf Grafik, Menüsteuerung und so weiter verzichtet.
Das abgedruckte Programmlisting soll dem interessierten Bastler lediglich als Stütze für eigene Projekte dienen.

Dieter Kinzinger/rs



Skizze 3: Anschluß der Temperaturmeßanlage an die PIO

Literatur:
JOYCE – mehr als ein Textsystem
DMV-VERLAG

Den JOYCE programmieren
Franzis Verlag

```
Program Temperaturmessung;
(*$I-*) (*$K-*) (*$R-) (*$U-*) (*$V-*)

(***********************************************)
(* TEMPX.PAS                                   *)
(* Joyce PCW 8256/8512                         *)
(* (c) 1992 Dieter Kinzinger & CPC International*)
(* Programm zur Temperaturmessung über das     *)
(* Schnittstellenmodul aus dem Buch: JOYCE -   *)
(* mehr als ein Textsystem.                    *)
(* Stand 27.06.92                              *)
(***********************************************)

Const  inp  = $A5;
Var    temp,mint,maxt,
       tempalt               : Integer;
       temp_str              : String[20];
       data                  : Array[1..200] Of Byte;
       d                     : Array[1..13] Of Byte;
       screen_on             : Boolean;

Procedure hidecursor;
Begin
 Write(#27,'f');
End;

Procedure Showcursor;
Begin
 Write(#27,'e');
End;

Function key : Boolean;   (*Prüft Tastaturstatus*)
Const bios_userf   = $FC5A;  (* nach CPC 2/3 92 *)
      kmget_status = $115C;
Var   A            : Byte;

Begin
 InLine ($CD/bios_userf/
         kmget_status/
         $32/A);
 If A = $FF Then key := True Else key := False;
End;

Procedure ende;
Begin
 ClrScr; GotoXY(35,20);
 WriteLN('Programm beendet');
 showcursor; halt;
End;

Procedure Messen;
Var n, totzeit : Integer;
Begin
 totzeit := 30000;
 n := 0;
 InLine($F3);                          (* DI *)
 Repeat
  n := n+1;
 Until (port[inp] Or 251 = 251) Or (n>totzeit);
 For n := 1 To 200 Do
  data[n] := port[inp] or 249;
 InLine($FB);                          (* EI *)
End;
```

```
Procedure auswertung;
Var dy, q, z : Integer;
Begin
 q := 1;
 For z := 1 To 13 Do
 Begin
  While ((data[q] Or 251)>251) And (q<199) Do
                                    q:=q+1;
  dy := (data[q] And 2);
  While ((data[q] Or 251)<255) And (q<199) Do
                                    q:=q+1;
  If dy = 2 Then d[z] := 0 Else d[z] := 1;
 End;
 temp := D[13] + D[12]*2 + D[11]*4 + D[10]*8 +
         (D[9] + D[8]*2 + D[7]*4 + D[6]*8)*10 +
         (D[5] + D[4]*2 + D[3]*4 + D[2]*8)*100;
 If D[1] = 1 Then temp := temp * (-1);
End;

Procedure mini_max;
Begin
 If temp>maxt Then maxt := temp;
 If temp<mint Then mint := temp;
End;

Procedure tempout(t : Integer);
Var s : String[4];
    l : Integer;
Begin
 temp_str := '';
 str(t,s);
 l := Length(s);
 Insert('.',s,l);
 temp_str := s+' Grad Cels.';
End;

Procedure digitemp;
Var p          : Byte;
    first,
    cancel     : Boolean;
Begin
 first := True;
 ClrScr;
 hidecursor;
 Repeat
  For p := 1 To 2 Do messen;
  cancel := key;
  auswertung;
  tempout(temp);
  GotoXY(30,12); Write('Temperatur: ',temp_str);
  If first Then Begin first:=False;mint:=temp;
                      maxt:=temp;
                End Else mini_max;
  tempout(mint);
  GotoXY(30,14);Write('Minimum    : ',temp_str);
  tempout(maxt);
  GotoXY(30,16);Write('Maximum    : ',temp_str);
 Until cancel;
end;

Begin
 digitemp;
 ende;
End.
```

PCW in neuem Gewand

PCW 9256: Klassiker in überarbeiteter Aufmachung

Obwohl schon seit langem angekündigt, wird erst jetzt der neue PCW auf dem deutschen Markt eingeführt. Spritziges Design, 3,5-Zoll-Laufwerk und volle Softwarekompatibilität zu den älteren Geschwistern zeichnen dieses Gerät aus.

"Wie bekomme ich nur meine Daten vom PCW auf meinen Laptop und wieder zurück?", hörte man zeitweise jeden Dienstag zur Hotline unsere Leser fragen. "200 DM müßten Sie schon mindestens investieren", so unsere Antwort. Ist es doch für den Transport der Daten zwischen PCW und PC erforderlich, entweder eine serielle Schnittstelle oder aber ein zweites Laufwerk im 3,5-Zoll-Format anzuschaffen. Den PCW zu Hause und einen Laptop unterwegs einzusetzen, ist dann kein Problem mehr, aber die 200,– DM bloß für den simplen Datenaustausch zwischen beiden Geräten stören doch ein wenig. Warum also nicht vor der Anschaffung eines PCW schon einmal über das verwendete Diskettenformat nachdenken?

Amstrad auf Abwegen?

Die Firma Amstrad ist jetzt dem Markt ein wenig hinterhergehopst und bietet auch einen PCW mit 3,5-Zoll-Laufwerk an. Das soll aber nicht die einzige Änderung bleiben!
Mit einem 720-kByte-Laufwerk als LW-A hat man sicher einen immensen Schritt nach vorne gemacht. Weil die CF2DD-Disketten im 3-Zoll-Format in letzter Zeit nur noch teuer oder gar nicht mehr zu bekommen waren, mußte der PCW-9512-Benutzer auf (noch) handelsübliche CF2-Disketten umsteigen. Ein arges Risiko. Beachte man doch die vorgesehene Formatierungsdichte von maximal 200 kByte pro Diskettenseite. Diskettenfehler sind also bei Benutzung des 720 KByte-Formates auf Dauer vorprogrammiert.
3,5-Zoll-Disketten sind dagegen schon wesentlich besser zu bekommen, sind sie doch im PC-Bereich zum Standard geworden. Hat man auf beiden Seiten – sowohl beim PCW als auch beim PC – ein 3,5-Zoll-Diskettenlaufwerk, muß weiterhin nur noch eine Transfersoftware, zum Beispiel TREAD, angeschafft werden. Daß dies wesentlich preisgünstiger ist als die zwei zuvor genannten Alternativen, ist klar.



Futuristisch präsentiert sich der PCW 9256. Sowohl Computer und Tastatur als auch der Drucker weisen ein elegantes Bogendesign auf

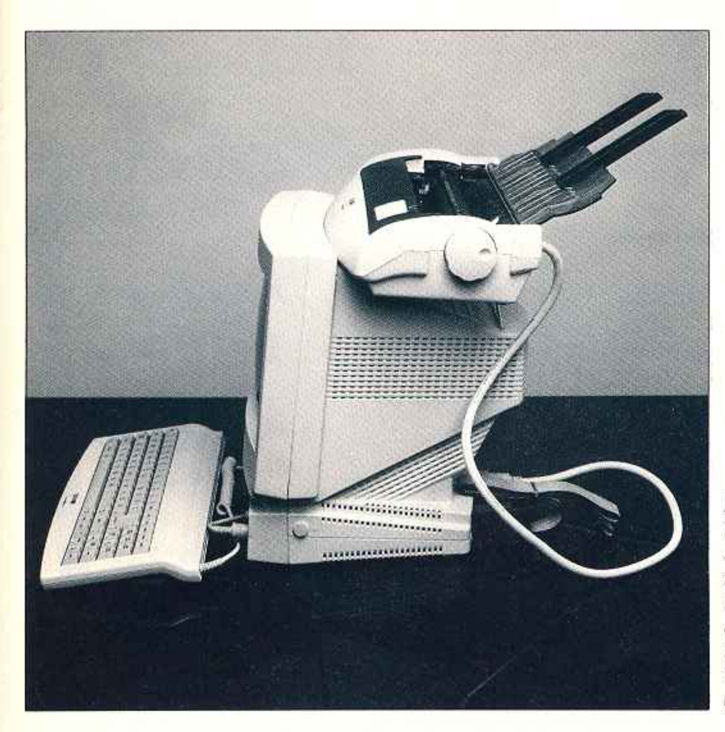
Der PCW9256 sowie die im Artikel angesprochene Hardware und Software wurde uns freundlicherweise von der Firma Wiedmann Unternehmensberatung, Korbiniansplatz 2, 8045 Ismaning, zur Verfügung gestellt.

Neuer Look

Der PCW 9256 ist jedoch nicht nur eine Kopie des PCW 9512 mit eingebautem 3,5-Zoll-Laufwerk, sondern weist noch weitere Änderungen auf. Auffälligstes Merkmal ist hier schon allein das Aussehen. Auch wenn der PCW 9512 schon futuristisch designed war, konnte nichts über sein plumpes Auftreten und seine großen Ausmaße hinwegtäuschen. Das 9256er Modell hingegen weiß durch geringe Ausmaße zu gefallen. So findet der schon als zierlich zu bezeichnende Joyce sicher auf jedem Schreibtisch seinen Platz. Den Drucker im Huckepack und die Tastatur, die sich bei Bedarf auch auf den "PCW-Turm" legen läßt, sind ebenfalls ein enormer Fortschritt gegenüber den älteren Modellen. Einen Rückschritt hat man mit Sicherheit in der Speicherausstattung gemacht. War der 9512 schon mit 512 kByte Arbeitsspeicher ausgestattet, gibt man den 9256ern nur noch 256 kByte mit. Der Speicher läßt sich zwar intern mit Speicherchips oder extern per SCA-RAMPack aufrüsten, trotzdem sollten mindestens 512 kByte von Haus aus drin sein. Muß man doch ohne die zusätzlichen 256 kByte auf so manches verzichten. Will man zum Beispiel LocoScript 2.32 mit Zusatzprogrammen einsetzen, ist das nicht uneingeschränkt möglich. Noch während des Bootens würde sich der PCW mit der Meldung "Speicher voll" verabschieden. Ebenso das Programm MicroDesign III, das bei 256 kByte gar nicht erst laufen möchte.

LocoScript 1.5 – wozu?

Neben Rechner, Tastatur und Drucker befindet sich noch Software im Lieferumfang des PCW. So aus unerfindlichen Gründen die LocoScript-Version 1.5. Eine Asbach-uralt-Version also, und dazu noch in Englisch. Aber optio-



Der Drucker wurde so konzipiert, daß er nicht nur neben, sondern direkt auf dem PCW-Monitor Platz findet. Wird der Rechner gerade nicht benötigt, kann die Tastatur dem Beispiel des Druckers folgen

nal läßt sich ja auf die deutsche Version 2.32 umrüsten. Ebenso findet man im Joyce-Paket noch eine Diskette mit CP/M Plus. Schaut man sich deren Inhalt an, fällt auf den ersten Blick nichts Ungewöhnliches auf, oder doch: Die vom PCW bekannte Systemdatei, die bisher mit .EMS endete, wurde mit einem anderen Endbuchstaben versehen. Aus .EMS wurde einfach .EMT gemacht. Das bedeutet, daß sich die Entwickler des PCW sicherlich eine Menge Mühe gemacht und sogar die Ursystemroutinen umgestrickt haben.
Beim weiteren Betrachten des Disketteninhalts fällt eine Datei mit dem Namen RAMTEST.COM auf. Ihre Aufgabe besteht darin, den Speicher zu testen und – wenn vorhanden – eine angesteckte Speichererweiterung anzumelden.

Nichts für Amateure

Wird nämlich eine Speichererweiterung angeschlossen, heißt dies nicht, daß dieser Zusatzspeicher gleich erkannt wird. Haareraufend kamen auch wir erst später darauf, das Programm RAMTEST zu starten. Und siehe da, der Zusatzspeicher in Form der Pro9256-Schnittstelle wird seitdem unter CP/M und LocoScript richtig erkannt.
Selbstverständlich muß man für den kompakten Bau des PCW auch ein wenig in Kauf nehmen. War es beim 9512 schon schwer, an die Innereien zu kommen, um gegebenenfalls ein zweites Laufwerk nachzurüsten, sollte beim 9256 ganz von diesem Vorhaben abgesehen werden. Genau wie bei den neuen CPCs wurde hier zwar auch an der Platine und deren Größe gefeilt, der unproportionale Schrumpfvorgang des Gehäuses hingegen läßt für weitere Einbauten keinen Platz. Gerüchten zufolge sollte ja innerhalb des neuen PCW schon ein Platz für eine Festplatte vorgesehen sein. Dieser läßt sich jedoch nirgends im Gehäuse lokalisieren.

Fazit

Lediglich ein Kabel erinnert daran, daß irgendwo unter dem Platinenwust noch Platz für ein zweites Laufwerk sein muß. Den Einbau und die Auswahl des Laufwerks sollte man jedoch aus vorgenannten Gründen dem Fachmann überlassen. Dies kostet zwar ein paar Mark, ist jedoch noch billiger als die Reparatur des eventuell beschädigten Geräts.
Im großen und ganzen kann man diesen PCW-Neubau als gelungen bezeichnen. Bringt er doch neben dem 3.5-Zoll-Laufwerk auch noch jede Menge Platzersparnis. Und Platz, der ist bekanntlich auf jedem Schreibtisch knapp. Das Einsatzgebiet des PCW 9256 liegt eindeutig wieder im Bereich der Datenerfassung und -auswertung.
Für diesen Zweck reicht in vielen Fällen auch die englische Konfiguration, wie sie nach Bestellung des Grundgeräts vorliegt. Englische Textverarbeitung sowie die 82-Tasten-QWERTY-Tastatur können dann später noch auf deutsche Ansprüche umgerüstetet werden. Dies ist zwar etwas umständlich, anfangs jedoch aus Kostenfragen sicherlich eine gute Entscheidung.

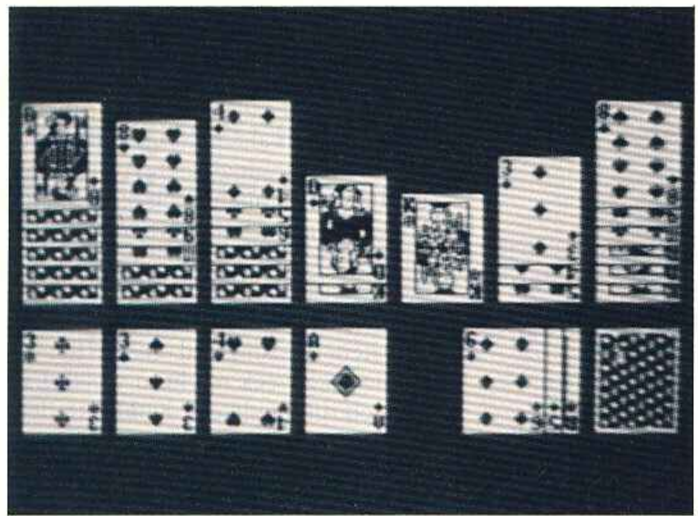
rs

Nach Aussage der Firma Wiedmann werden ab sofort alle Programmpakete auch für den PCW 9256 auf 3,5-Zoll-Disketten angeboten. Für den Test des PCW stand uns folgende Konfiguration zur Verfügung:
PCW 9256 mit einem Laufwerk, Systemdrucker und deutscher Tastatur, zusätzlich noch die Pro9256-Schnittstelle, die neben einem parallelen Port zum Anschluß eines externen Druckers noch 1 MByte Zusatzspeicher bereitstellt. Auf der Softwareseite lagen uns LocoScript 2.32 sowie das englische Programmpaket Cracker II und die neueste Version von MicroDesign vor. Damit sich problemlos Einzelblattpapier einlagen läßt, stand zusätzlich noch der Margin Maker MM4 zur Verfügung.

CPC International Druckertest: PCW 9256 Systemdrucker

Normale Schrift, Hochgestellt und Tiefgestellt
Normale Schrift, Hochgestellt und Tiefgestellt
Normale Schrift, Hochgestellt und Tiefgestellt
Normale Schrift, Hochgestellt und Tiefgestellt
Dies ist 9-Nadel-Briefqualität; Umlaute: äöüÄÖÜß
Dies ist 9-Nadel-Briefqualität; Umlaute: äöüÄÖÜß
Dies ist 9-Nadel-Briefqualität; Umlaute: äöüÄÖÜß
Dies ist 9-Nadel-Briefqualität; Umlaute: äöüÄÖÜß

Das Schriftbild des PCW-Druckers reicht für den normalen Schriftverkehr vollkommen aus



Spielkarten – ein uraltes Vergnügen. Die nächste Ausgabe bringt Kartenspiel-Spaß für Tüftler: Freuen Sie sich auf die grafisch ausgefeilte "Solitär"-Patience für den CPC

»CPC International«12'92/1'93 erhalten Sie nur im **Bahnhofsbuchhandel** oder **direkt beim Verlag** ab:



CPC

– Wir zeigen den PC-"Windowsianern", was eine britische 8-Bit-Harke ist! Nachdem wir in diesem Heft bereits eine hervorragende "Minesweeper"-Umsetzung gebracht haben, präsentieren wir im nächsten Heft die CPC-Adaption eines weiteren Windows-Spieleklassikers: Solitär! Die CPC-Version überzeugt mit wunderschöner Grafik und zeigt einmal mehr, was Amstrads "Kleine" auch heute noch können. Alle Freunde intelligenter Kombinationsspiele dürfen sich jetzt schon einmal die Lippen lecken.

– Fraktale Grafiken haben eine starke Anziehungskraft. Ihre "chaotische" Harmonie entspringt der geheimnisvollen Verbindung von realer und imaginärer Zahlenwelt, den Windungen immer wieder in sich selbst zurückgeführter Funktionen. Sie reizen Philosophen immer aufs Neue dazu, über das Woher und Wohin nachzudenken. Wir bringen eine der attraktivsten Fraktalgruppen auf Ihren CPC: die Baumfraktale!

– Die Rollenspiel-Serie geht ins prunkvolle Finale! Jetzt fügen sich alle vorgestellten Techniken zum spannenden und atmosphäregeladenen Spiel zusammen. Versäumen Sie nicht die Geburt Ihrer eigenen Heldendynastie!

– Dateien und ihr interner Aufbau sind für viele CPCler noch so etwas wie böhmische Dörfer. Dabei kann es speziell bei der Fehlersuche sehr hilfreich sein, wenn man sich mit Filetypen und Headern, Blöcken und Attributen auskennt. Unser Grundlagenartikel verschafft Ihnen Durchblick.

BERICHT

– Auf dem Software-Markt für die CPCs herrschen zwar nicht mehr die Wirbelstürme der Jahre 1986 bis 1988. Dennoch gibt es immer wieder mal eine frische Brise. Zum Beispiel mitten im Ruhrpott: Da macht ein neuer Software-Versand mit eigener Programmentwicklung auf, und der kümmert sich doch tatsächlich um CPCler! Wir haben die Firma Ralf Brostedt in Herne besucht und dabei einige Überraschungen erlebt.

– Was schenkt man einem eingefleischten CPC- oder PCW-Benutzer zu Weihnachten? Wir haben uns ein wenig umgesehen und sagen Ihnen, wo Sie nützliches Zubehör und lustige Gimmicks für die Computerei mit den kleinen Amstrads bekommen.

CP/M

– Public-Domain-Software ist eine extrem feine Sache: Man darf sie frei kopieren und weiterverbreiten. Zwar ist die Menge der PD-Programme für das Betriebssystem CP/M nicht mit dem Riesenpool für die IBM-kompatiblen PCs zu vergleichen. Dennoch gibt es bemerkenswerte Utilities, Programmiersprachen und Anwendungen in der Public Domain. Wir stellen Ihnen einige vor.

HARDWARE

– Was sind EPROMS eigentlich? Wie funktionieren sie, wie spricht man sie an, wofür kann man sie einsetzen? Wenn Sie sich für die kleinen programmierbaren "Festgedächtnisse" interessieren, ist unser Know-how-Beitrag in der nächsten Ausgabe genau das richtige für Sie.

PCW

– Grafik vom Feinsten bietet das englische Programm "Micro Design" bereits seit der ersten Version. Daß eine Steigerung dennoch möglich ist, beweist die aktuelle Version 3, die gerade taufrisch auf unserem Redaktionstisch gelandet ist. Ausgefeilte Text- und Kurvenfunktionen machen die Arbeit mit "Micro Design 3" noch angenehmer und lassen die Ergebnisse noch professioneller aussehen als bei den Vorgängerversionen.

DATABOX

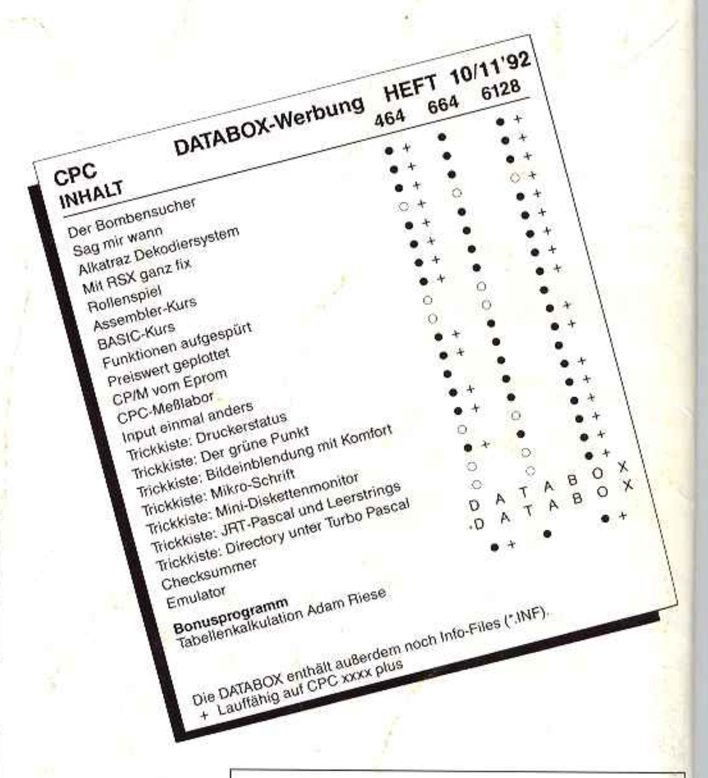


Für alle CPCs als Kassette und Diskette.

Das ist die Software zur CPC INTERNATIONAL Jeden Monat neu

DATABOX:

- mehr als der übliche Software-Service
- bringt ergänzend sämtliche Listings der jeweiligen Zeitschrift und alle Programmbeispiele auf Kassette oder auf Diskette
- Die Programme sind, soweit systembedingt möglich, auf allen drei CPC-Modellen lauffähig. Einzelheiten entnehmen Sie bitte der nebenstehenden Aufstellung.
- Soweit die Programme nicht Bestandteil einer Serie sind, befinden sich alle Programme als »ready to run« auf der DATABOX.
- erscheint jeden Monat und trägt das Titelbild des gleichzeitig erscheinenden Heftes
- Der Datenträger zur CPC International enthält außerdem jedesmal ein zusätzliches Bonusprogramm, das nicht im Heft abgedruckt ist.



DATABOX-Werbung HEFT 10/11'92

CPC INHALT	464	664	6128
Der Bombensucher	● +	●	● +
Sag mir wann	● +	●	● +
Alkatraz Dekodiersystem	● +	●	● +
Mit RSX ganz fix	○ +	○	○ +
Rollenspiel	● +	●	● +
Assembler-Kurs	● +	●	● +
BASIC-Kurs	● +	●	● +
Funktionen aufgespürt	● +	●	● +
Preiswert geplottet	○	○	● +
CP/M vom Eprom	○	○	●
CPC-Meßlabor	● +	●	● +
Input einmal anders	● +	●	● +
Trickkiste: Druckerstatus	●	●	●
Trickkiste: Der grüne Punkt	● +	●	● +
Trickkiste: Bildeinblendung mit Komfort	● +	●	● +
Trickkiste: Mikro-Schrift	○	○	● +
Trickkiste: Mini-Diskettenmonitor	● +	●	● +
Trickkiste: JRT-Pascal und Leerstrings	○	○	● +
Trickkiste: Directory unter Turbo Pascal	○	○	● +
Checksummer	D A	T A B	O X
Emulator	D A	T A B	O X
Bonusprogramm Tabellenkalkulation Adam Riese	● +	●	● +

Die DATABOX enthält außerdem noch Info-Files (*.INF).
+ Lauffähig auf CPC xxxx plus

Einzelbezugspreise für DATABOX:

3-Zoll-Diskette

Inland:		**Ausland:**	
Einzelpreis	24,– DM	Einzelpreis	24,– DM
zzgl. Versandkosten	4,– DM	zzgl. Versandkosten	6,– DM
Endpreis	**28,– DM**	**Endpreis**	**30,– DM**

5 1/4-Zoll-Diskette

Inland:		**Ausland:**	
Einzelpreis	21,– DM	Einzelpreis	21,– DM
zzgl. Versandkosten	4,– DM	zzgl. Versandkosten	6,– DM
Endpreis	**25,– DM**	**Endpreis**	**27,– DM**

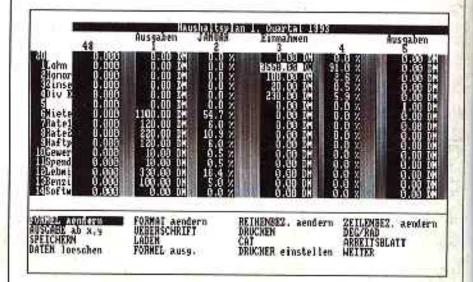
Kassette

Inland:		**Ausland:**	
Einzelpreis	14,– DM	Einzelpreis	14,– DM
zzgl. Versandkosten	4,– DM	zzgl. Versandkosten	6,– DM
Endpreis	**18,– DM**	**Endpreis**	**20,– DM**

Zahlungsweise:
am einfachsten per Vorkasse (Verrechnungsscheck) oder als Nachnahme zuzüglich der Nachnahmegebühr. Bei Lieferungen ins Ausland ist Nachnahme nicht möglich.

Bitte benutzen Sie die Bestellkarte.

Nur auf der Databox



- **Adam Riese: Professionelle Tabellenkalkulation (Siehe Seite 47)**
- **Rollenspiel-Kurs: Komplexe Landschafts- und Spielergrafikdateien**
- **Sag mir wann: Geschichtsdatei mit ca. 100 Ereignissen**
- **CP/M vom Eprom: Infos und Programmanpassung für Sonderfälle**

DMV-Verlag · Postfach 250 · W-3440 Eschwege

